Orange Pi R2S
User Manual

Catalogue

1. Basic characteristics of Orange Pi R2S

1. 1. What is Orange Pi R2S

The Orange Pi R2S adopts an open chip micro Ky X1 8-core RISC-V AI processor, providing universal computing power with 2TOPS CPU fusion. It has 2GB/4GB/8GB LPDDR4X operating memory and supports 8GB eMMC.

Orange Pi R2S has a wide range of interfaces, including 2.5G high-speed dual network ports, dual gigabit network interfaces, USB 2.0, USB 3.0, Type-C power supply, TF card slot, debugging serial port, etc.

Orange Pi R2S is exquisite and compact, with a size of only 79.2*46*1.6 mm, making it easy to install in various small spaces. Orange Pi R2S supports OpenWrt and Ubuntu, and can be widely used in enterprise gateways, industrial control hosts, industrial automation, energy management, smart transportation, smart cities, and other fields.

1. 2. Purpose of Orange Pi R2S

We can use it to achieve:

- A router
- One switch

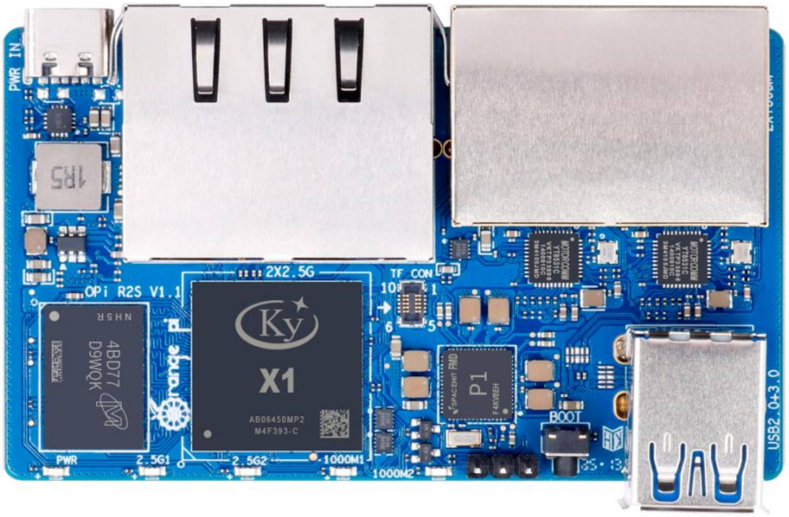
Of course, there are many other features as well. With a powerful ecosystem and various expansion accessories, Orange Pi can help users easily achieve delivery from creativity to prototype to mass production. It is an ideal creative platform for makers, dreamers, and hobbyists.

1. 3. Hardware Features of Orange Pi R2S

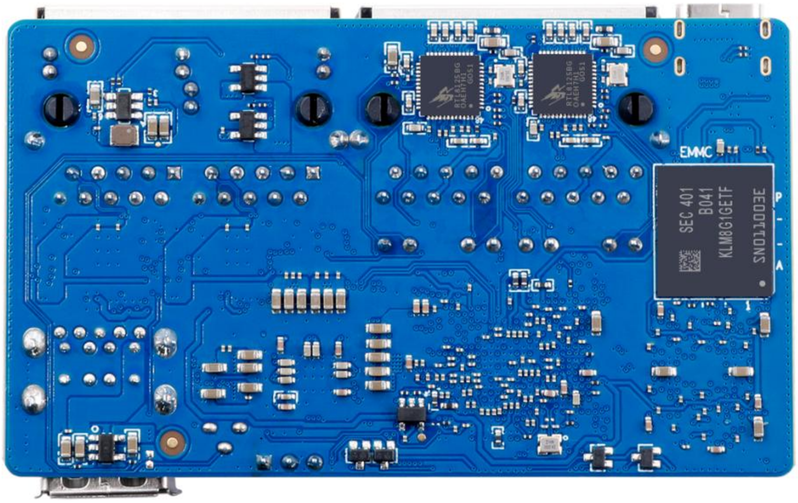
Introduction to Hardware Features	
CPU	• Ky X1 8 Core RISC-V AI • 2.0T computing power
Memory	2GB/4GB/8GB（LPDDR4X）
Onboard storage	• 8GB eMMC
Ethernet	2 * Gigabit Ethernet port（YT8531C) 2 * PCIe 2.5G Ethernet ports（RTL8125BG)
USB interface	• 1 * USB 2.0 supports Device or HOST mode • 1 * USB3.0 HOST
Debug UART	3 PIN debugging serial port
LED lamp	* Power light, 4 * Ethernet port light
key	1 * BOOT key
power supply	Type-C interface power supply 5V/3A
Supported operating systems	Operating systems such as Ubuntu and OpenWRT
Introduction to appearance specifications	
Product size	79.2*46*1.6 mm
Weight	60g
range Pi™ is a registered trademark of Shenzhen Xunlong Software Co., Ltd.	

1. 4. Top and Bottom Views of Orange Pi R2S

Top level view:

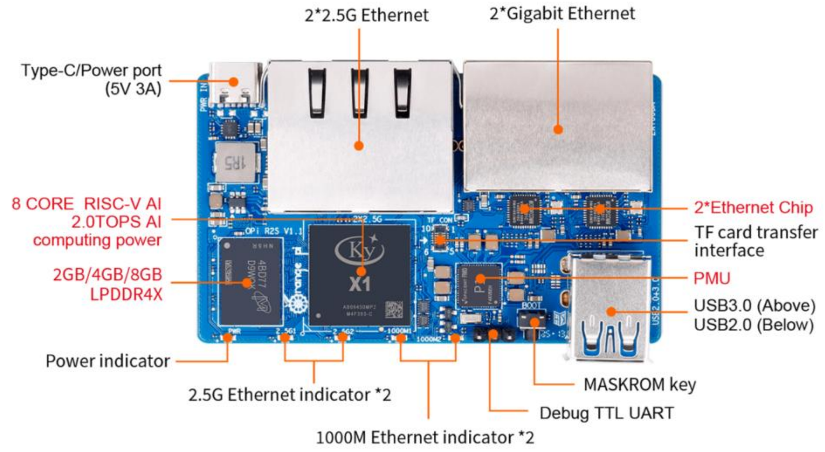


Bottom level view:

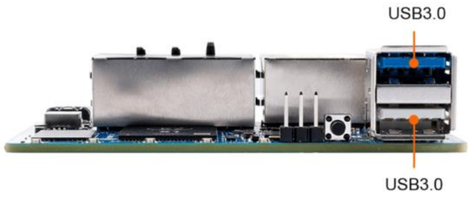


1.5. Interface details diagram of Orange Pi R2S

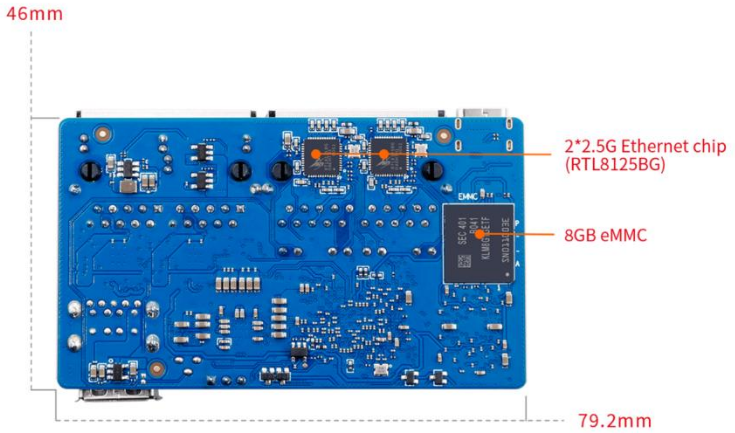
Product display



Top View



Side View



Bottom View

2. Introduction to using the development board

2. 1. Prepare the necessary accessories

1) TF card, a high-speed flash card with a minimum capacity of 16GB (recommended 32GB or above) and **class10** or above.

2) TF card reader, used to burn images onto TF cards.

3) For the Orange Pi R2S power adapter, it is recommended to use a 5V3A Type-C power supply.

The Type-C power interface of the development board does not support PD negotiation function and only supports a fixed 5V voltage input.

4) 100Mbps or 1G Ethernet cable, used to connect the development board to the Internet.

5) USB 2.0 male to male data cable, used for burning image function.

6) When using the serial port debugging function, a **3.3V** USB to TTL module and DuPont cable are required to connect the development board and computer.

7) A personal computer with Ubuntu and Windows operating systems installed.

1	Ubuntu22.04 PC	Optional, used for compiling Linux source code
2	Windows PC	Used for burning Linux images

2. 2. Download the image of the development board and related materials

1) The download link for the English version of the material is:

http://www.orangepi.org/html/hardWare/computerAndMicrocontrollers/details/Orange-Pi-R2S.html

2) The information mainly includes

a. **Linux source code**： saved on Github.

b. **User manual and schematic**： saved on Google Cloud Drive

c. **Official tools**： This mainly includes the software required during the use of the development board.

d. **Ubuntu image**： saved on Google Cloud Drive

e. **OpenWRT image**： saved on Google Cloud Drive

2. 3. Method of burning Linux image to TF card based on Windows PC

Note that the Linux image referred to here specifically refers to Linux distribution images such as Ubuntu and OpenWRT downloaded from the Orange Pi data download page.

2. 3. 1. Method of burning Linux images using balenaEtcher

1) Firstly, prepare a 16GB or larger TF card with a transfer speed of **class10** or above. It is recommended to use TF cards from brands such as SanDisk.

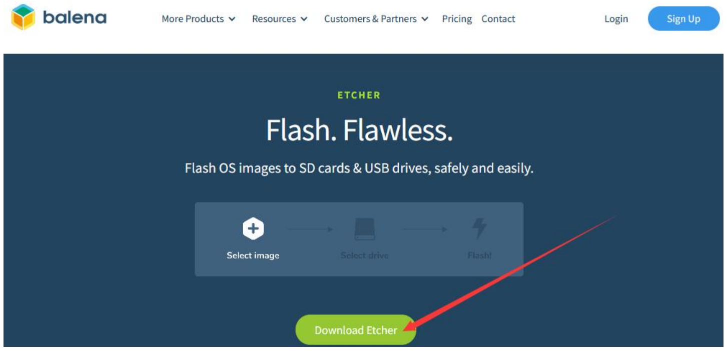
2) Then use a card reader to insert the TF card into the computer.

3) Download the compressed file of the Linux operating system image that you want to burn from the **Orange Pi's download page**, and then use decompression software to decompress it. In the decompressed file, the file ending with "**.img**" is the operating system image file, which is usually over 2GB in size.

4) Then download the Linux image burning software -- **balenaEtcher**, from the following download link:

https://www.balena.io/etcher/

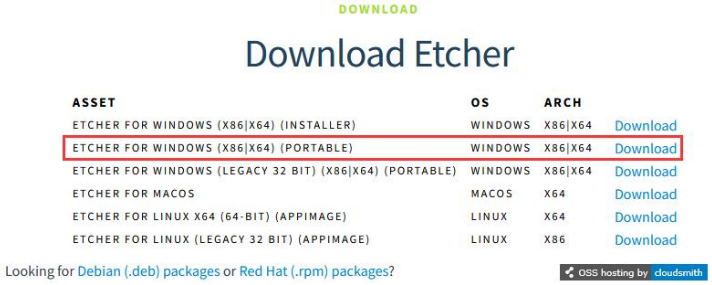
5) After entering the balenaEtcher download page, clicking the green download button will jump to the software download location.



Flash. Flawless.

Flash OS images to SD cards & USB drives, safely and easily.

6) Then you can choose to download the Portable version of BalenaEtcher software, which does not require installation and can be used by double clicking.



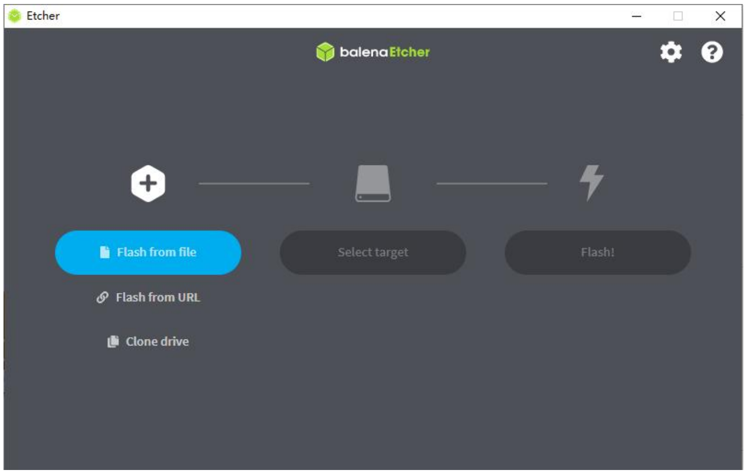
DOWNLOAD

Download Etcher

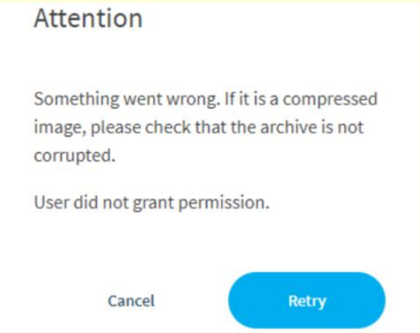
ASSET	OS	ARCH	
ETCHER FOR WINDOWS (X86|X64) (INSTALLER)	WINDOWS	X86|X64	Download
ETCHER FOR WINDOWS (X86|X64) (PORTABLE)	WINDOWS	X86|X64	Download
ETCHER FOR WINDOWS (LEGACY 32 BIT) (X86|X64) (PORTABLE)	WINDOWS	X86|X64	Download
ETCHER FOR MACOS	MACOS	X64	Download
ETCHER FOR LINUX X64 (64-BIT) (APPIMAGE)	LINUX	X64	Download
ETCHER FOR LINUX (LEGACY 32 BIT) (APPIMAGE)	LINUX	X86	Download

Looking for Debian (.deb) packages or Red Hat (.rpm) packages?

7) If you are downloading a version of balenaEtcher that requires installation, please install it first before using it. If you download the Portable version of balenaEtcher, simply double-click to open it. The interface of balenaEtcher after opening is shown in the following figure:



When opening balenaEtcher, if prompted with the following error:

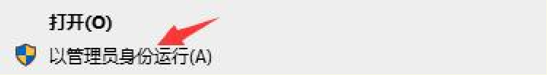


Attention

Something went wrong. If it is a compressed image, please check that the archive is not corrupted.

User did not grant permission.

Please select balenaEtcher and right-click, then choose to run as administrator.

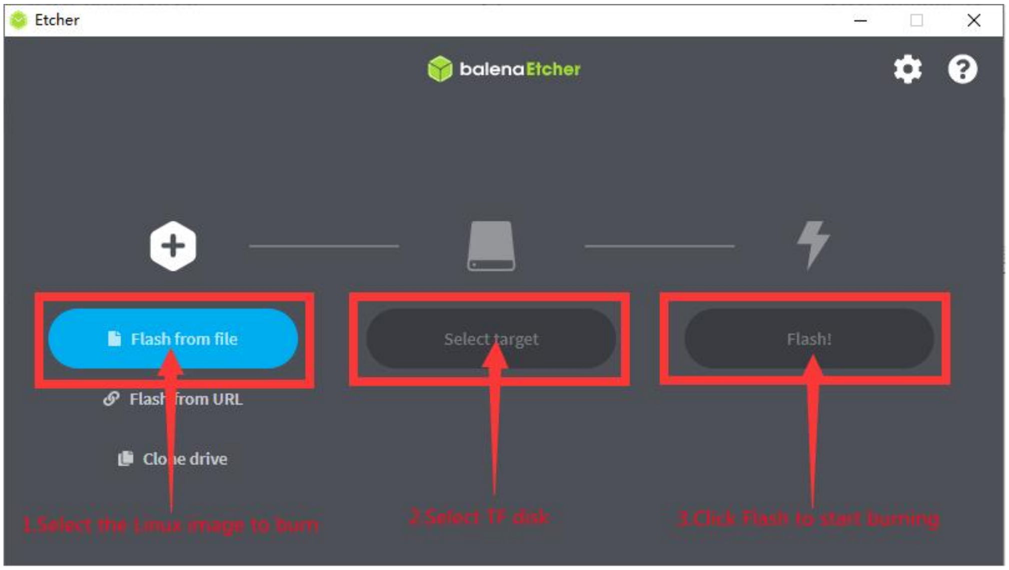


8) The specific steps for burning a Linux image using balenaEtcher are as follows:

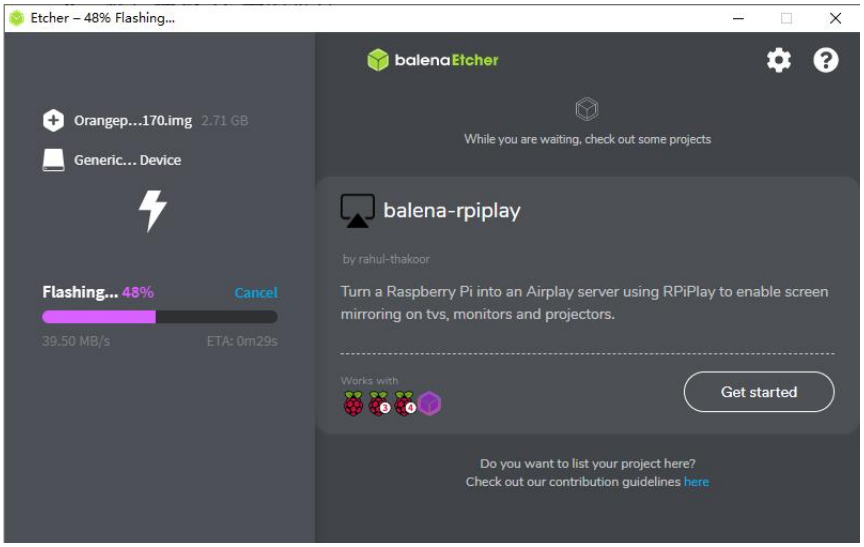
a. Firstly, select the path of the Linux image file to be burned.

b. Then select the drive letter of the TF card.

c. Finally, clicking Flash will start burning the Linux image onto the TF card.



9) The interface displayed during the process of burning a Linux image by balenaEtcher is shown in the following figure. In addition, the progress bar displaying purple indicates that the Linux image is being burned to the TF card.

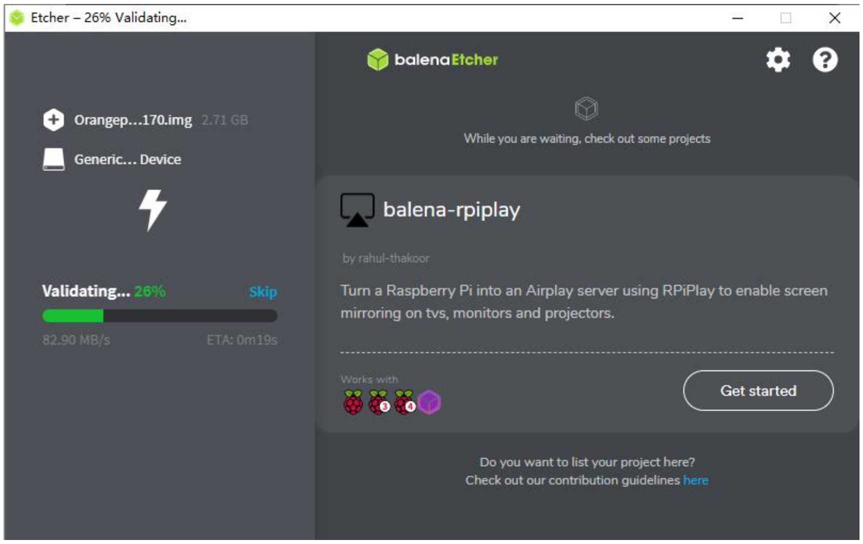


While you are waiting, check out some projects

Turn a Raspberry Pi into an Airplay server using RPiPlay to enable screen mirroring on tvs, monitors and projectors.

Do you want to list your project here?
Check out our contribution guidelines here

10) After the Linux image is burned, balenaEtcher will also verify the image burned to the TF card by default to ensure that there are no problems during the burning process. As shown in the following figure, a green progress bar indicates that the image has been burned and balenaEtcher is verifying the burned image.

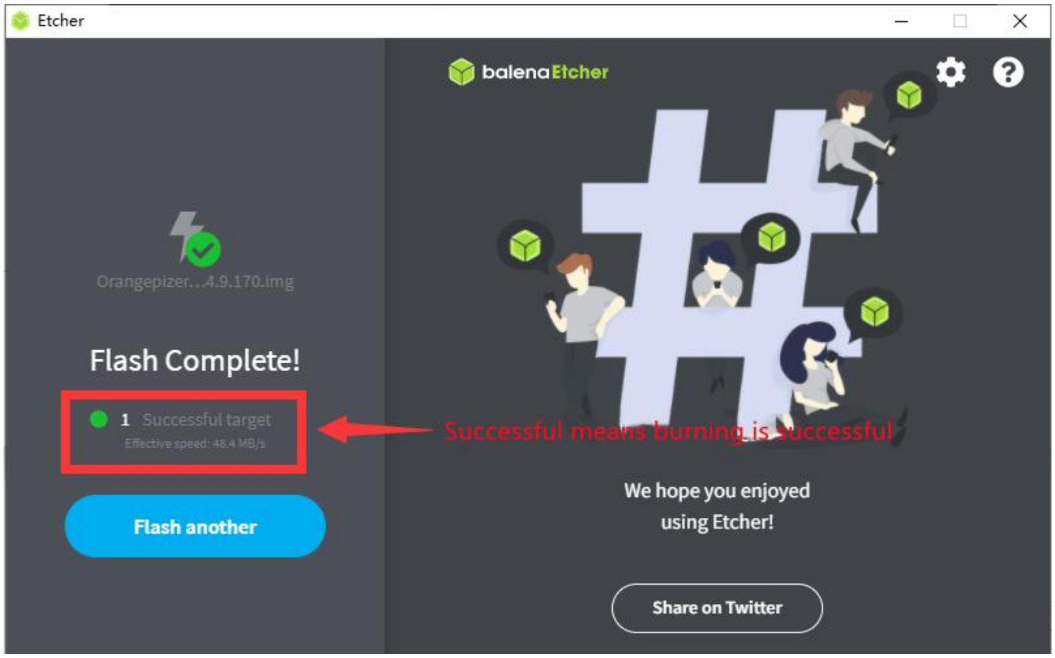


While you are waiting, check out some projects

Turn a Raspberry Pi into an Airplay server using RPiPlay to enable screen mirroring on tvs, monitors and projectors.

Do you want to list your project here?
Check out our contribution guidelines here

11) After successful burning, the display interface of balenaEtcher is shown in the following figure. If a green indicator icon is displayed, it indicates that the image burning is successful. At this time, you can exit balenaEtcher, then unplug the TF card and insert it into the TF card slot of the development board for use.


Etcher
balenaEtcher
Orangepizer...4.9.170.img
Flash Complete!
1 Successful target
Effective speed: 48.4 MB/s
Successful means burning is successful
Flash another
We hope you enjoyed using Etcher!
Share on Twitter

2. 4. Method for burning Linux images to TF cards based on Ubuntu PC

Note that the Linux image referred to here specifically refers to Linux distribution images such as Ubuntu and OpenWRT downloaded from the Orange Pi data download page, while Ubuntu PC refers to a personal computer with the Ubuntu system installed.

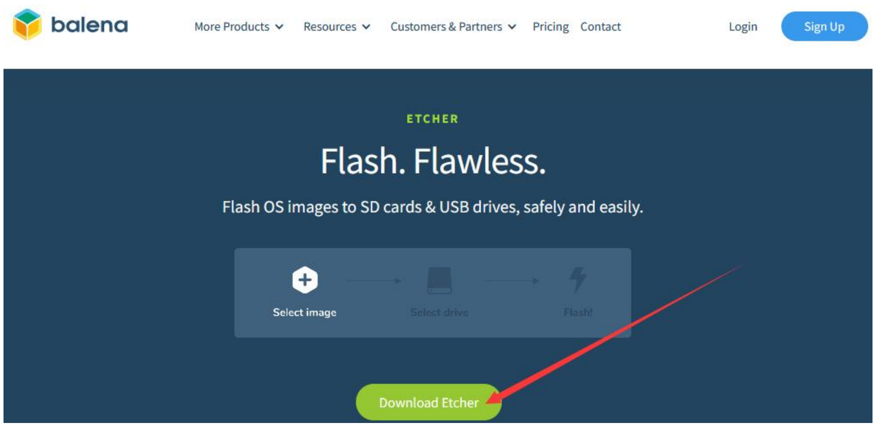
1) Firstly, prepare a 16GB or larger TF card with a transfer speed of **class10** or above. It is recommended to use TF cards from brands such as SanDisk.

2) Then use a card reader to insert the TF card into the computer.

3) Download the balenaEtcher software from the following link:

```
https://www.balena.io/etcher/
```

4) After entering the balenaEtcher download page, clicking the green download button will jump to the software download location.

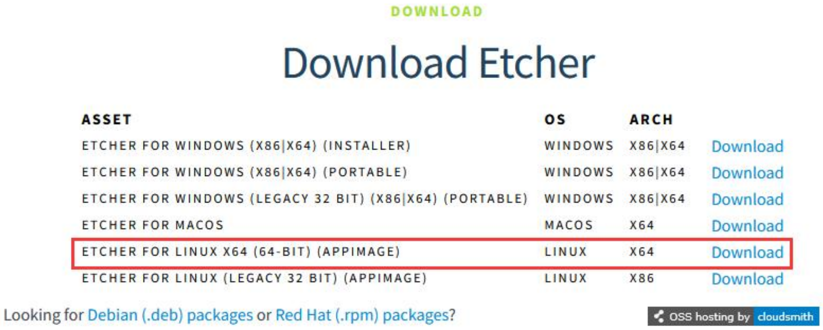


ETCHER

Flash. Flawless.

Flash OS images to SD cards & USB drives, safely and easily.

5) Then choose to download the Linux version of the software.



DOWNLOAD

Download Etcher

ASSET	OS	ARCH	
ETCHER FOR WINDOWS (X86|X64) (INSTALLER)	WINDOWS	X86|X64	Download
ETCHER FOR WINDOWS (X86|X64) (PORTABLE)	WINDOWS	X86|X64	Download
ETCHER FOR WINDOWS (LEGACY 32 BIT) (X86|X64) (PORTABLE)	WINDOWS	X86|X64	Download
ETCHER FOR MACOS	MACOS	X64	Download
ETCHER FOR LINUX X64 (64-BIT) (APPIMAGE)	LINUX	X64	Download
ETCHER FOR LINUX (LEGACY 32 BIT) (APPIMAGE)	LINUX	X86	Download

Looking for Debian (.deb) packages or Red Hat (.rpm) packages?

6) Download the compressed file of the Linux operating system image that you want to burn from the **Orange Pi's download page**, and then use decompression software to decompress it. In the decompressed file, the file ending with "**.img**" is the operating system image file, which is usually over 2GB in size.

The decompression command for the compressed file ending in 7z is as follows:

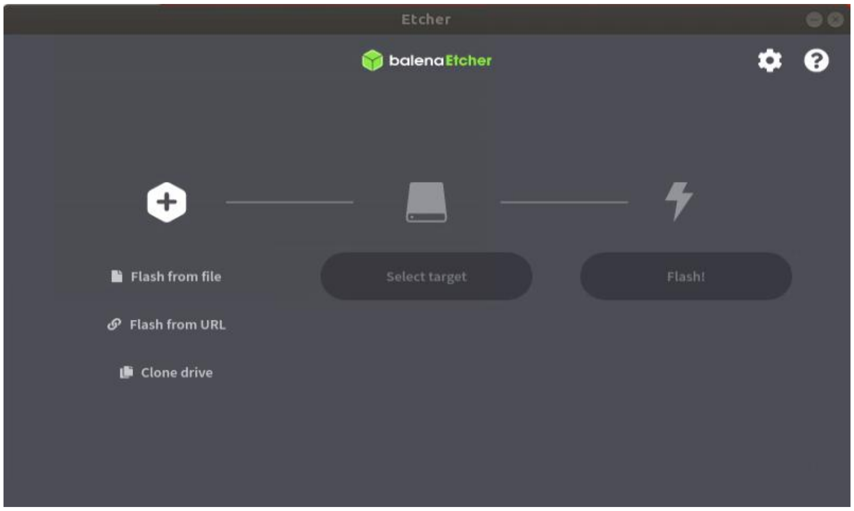
```
test@test:~$ 7z x Orangepir2s_1.0.0_ubuntu_noble_server_linux6.6.63.7z
test@test:~$ ls Orangepir2s_1.0.0_ubuntu_noble_server_linux6.6.63.*
Orangepir2s_1.0.0_ubuntu_noble_server_linux6.6.63.7z
Orangepir2s_1.0.0_ubuntu_noble_server_linux6.6.63.sha     #Verification and file
Orangepir2s_1.0.0_ubuntu_noble_server_linux6.6.63.img     #image file
```

7) After decompressing the image, you can first use the **sha256sum -c *.sha** command to calculate if the checksum is correct. If the prompt is **successful**, it means that the downloaded image is correct and can be safely burned to the TF card. If the prompt is that **the checksum does not match**, it means that there is a problem with the downloaded

image. Please try downloading it again.

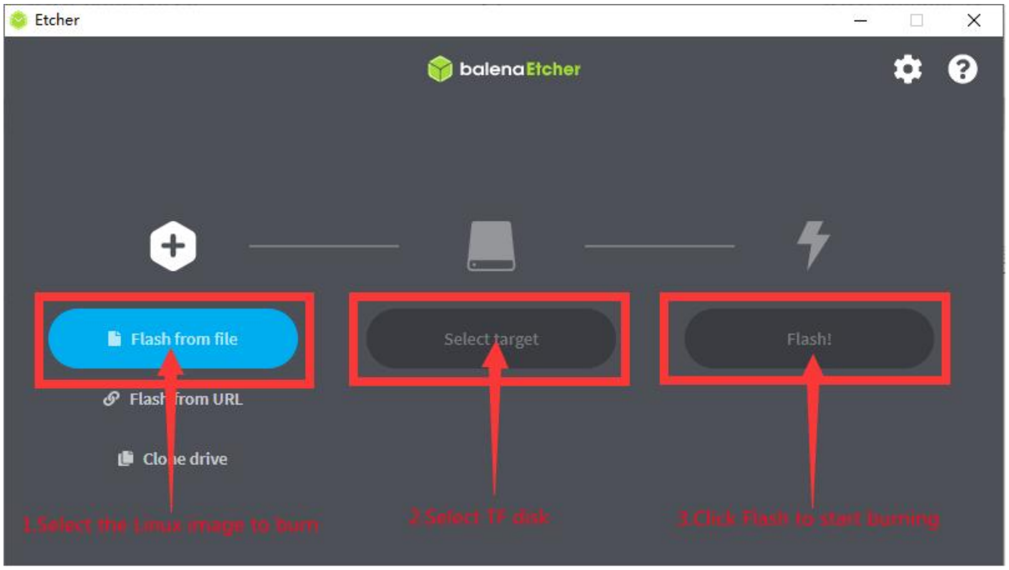
```
test@test:~$ sha256sum -c *.sha
Orangepir2s_1.0.0_ubuntu_noble_server_linux6.6.63.img: OK
```

8) Then double-click **balenaEtcher-1.5.109-x64.AppImage** on the graphical interface of Ubuntu PC to open BalenaEtcher (no installation required). The interface displayed after opening BalenaEtcher is shown in the following figure.

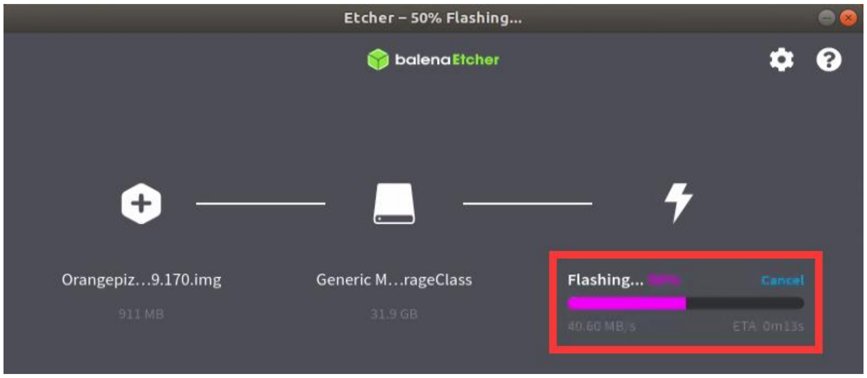


9) The specific steps for burning a Linux image using balenaEtcher are as follows:

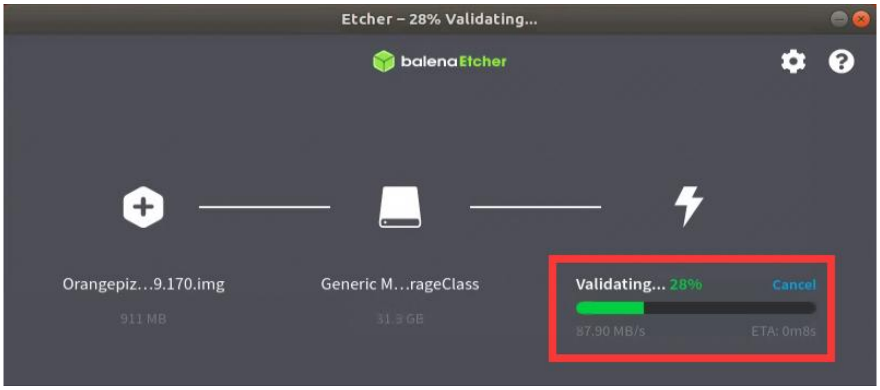
a. Firstly, select the path of the Linux image file to be burned.
b. Then select the drive letter of the TF card.
c. Finally, clicking Flash will start burning the Linux image onto the TF card.



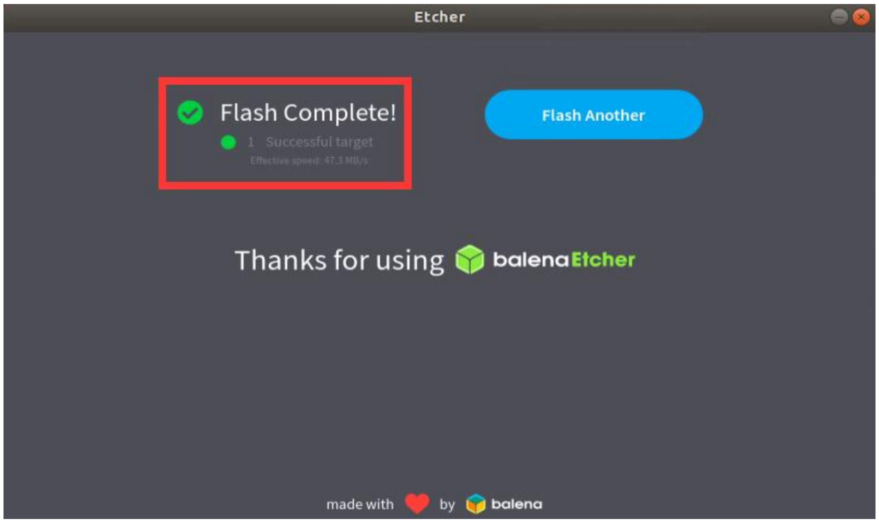
10) The interface displayed during the process of burning a Linux image by balenaEtcher is shown in the following figure. In addition, the progress bar displaying purple indicates that the Linux image is being burned to the TF card.



12) After the Linux image is burned, balenaEtcher will also verify the image burned to the TF card by default to ensure that there are no problems during the burning process. As shown in the following figure, a green progress bar indicates that the image has been burned and balenaEtcher is verifying the burned image.



13) After successful burning, the display interface of balenaEtcher is shown in the following figure. If a green indicator icon is displayed, it indicates that the image burning is successful. At this time, you can exit balenaEtcher, then unplug the TF card and insert it into the TF card slot of the development board for use.

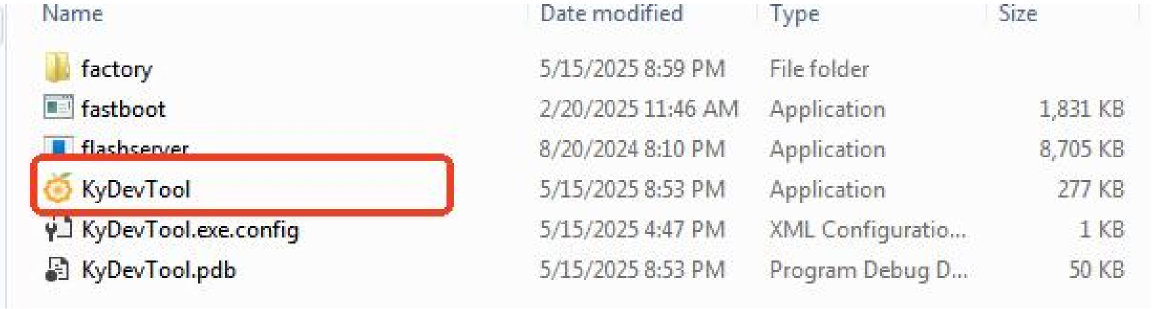


2. 5. Method for burning Linux images to eMMC

2. 5. 1. Method of burning Linux images to eMMC using KyDevTool tool

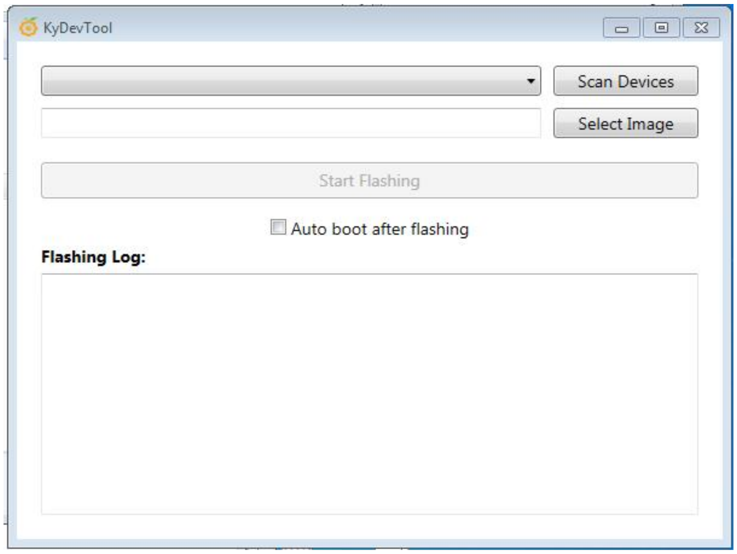
1）Firstly, it is necessary to prepare a high-quality USB 2.0 male to male data cable

2）Then download the **KyDevTool.zip** tool from the **Orange Pi's download page**, and open **KyDevTool.exe** in the decompressed folder.



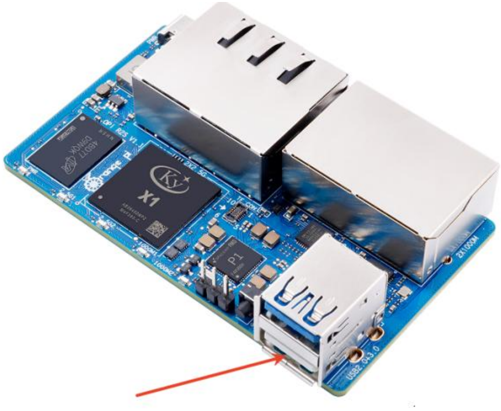
Name	Date modified	Type	Size
factory	5/15/2025 8:59 PM	File folder	
fastboot	2/20/2025 11:46 AM	Application	1,831 KB
flashserver	8/20/2024 8:10 PM	Application	8,705 KB
KyDevTool	5/15/2025 8:53 PM	Application	277 KB
KyDevTool.exe.config	5/15/2025 4:47 PM	XML Configuratio...	1 KB
KyDevTool.pdb	5/15/2025 8:53 PM	Program Debug D...	50 KB

3）The interface of **KyDevTool** tool is shown in the following figure.



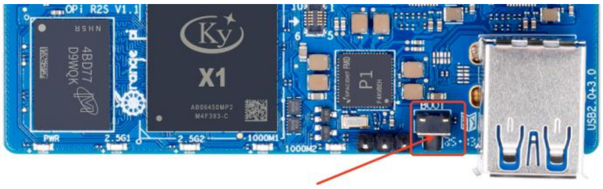
4）Then start burning the image to eMMC.

a. Firstly, connect the development board to the Windows computer via a USB 2.0 male to female data cable. The location of the USB 2.0 burning interface on the development board is shown in the following figure.



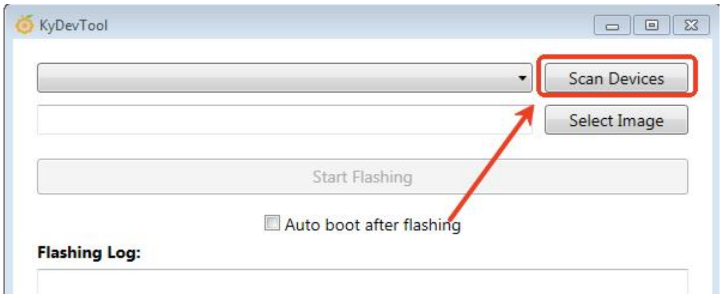
b. Then ensure that the development board is not connected to a power source.

c. Then hold down the BOOT button on the development board. The BOOT button is located on the development board as shown in the following figure:

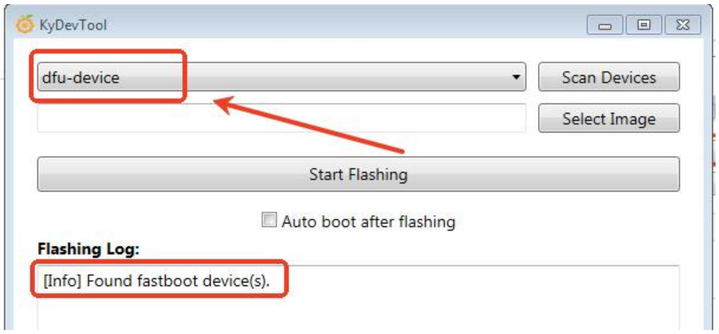


d. Then connect the Type-C interface power supply to the development board and power it on.

e. Then click on the **Scan Devices** button of the **KyDevTool** tool.

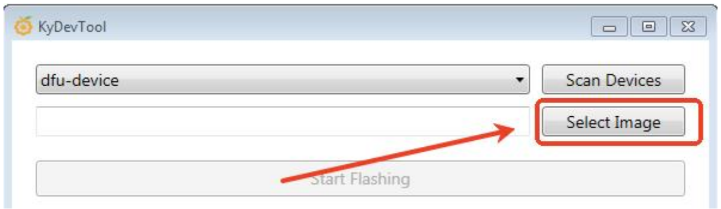


f. If everything is normal, a **fastboot** device should be found.

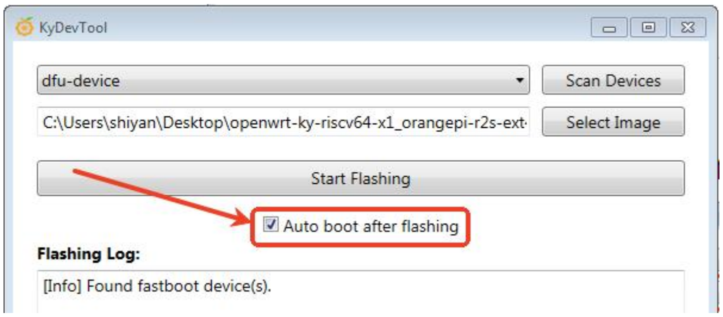


```
[Info] Found fastboot device(s).
```

g. Then click the **Select Image** button and choose the path of the Linux image you want to burn.



h. Then check the checkbox shown in the figure below, so that the burning system will automatically start after completion.

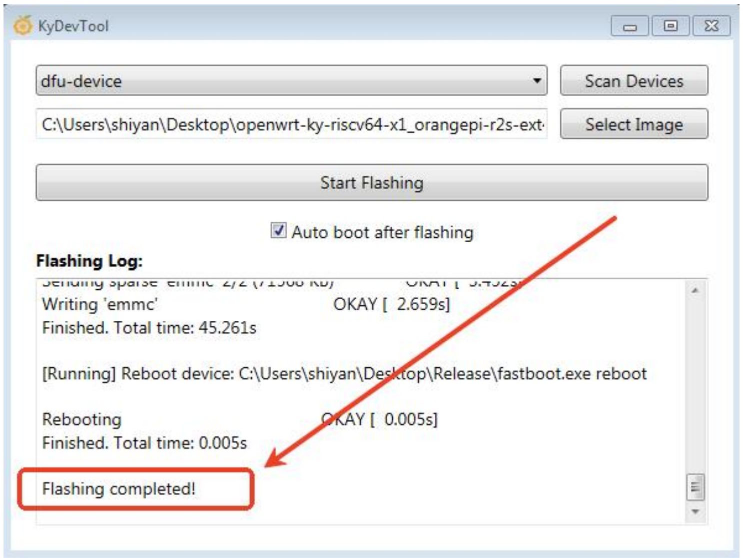

KyDevTool
dfu-device
Scan Devices
C:\Users\shiyan\Desktop\openwrt-ky-riscv64-x1_orangepi-r2s-ext
Select Image
Start Flashing

☑ Auto boot after flashing

Flashing Log:

```
[Info] Found fastboot device(s).
```

i. Finally, click the **Start Flashing** button to start burning the image, and the printed image after burning is shown in the following figure.


KyDevTool
dfu-device
Scan Devices
C:\Users\shiyan\Desktop\openwrt-ky-riscv64-x1_orangepi-r2s-ext
Select Image
Start Flashing

☑ Auto boot after flashing

Flashing Log:

```
Sending sparse 'emmc' 2/2 (71508 KB)              OKAY [  3.452s]
Writing 'emmc'                                    OKAY [  2.659s]
Finished. Total time: 45.261s

[Running] Reboot device: C:\Users\shiyan\Desktop\Release\fastboot.exe reboot

Rebooting                                         OKAY [  0.005s]
Finished. Total time: 0.005s

Flashing completed!
```

2. 6. Method for burning Linux images to USB storage devices

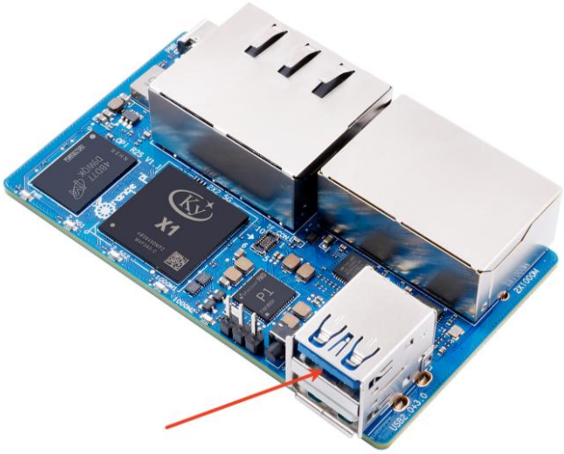
Note that the Linux image referred to here specifically refers to Linux distribution images such as Ubuntu and OpenWRT downloaded from the Orange Pi data download page.

1) eMMC comes pre installed with Linux system at the factory, so it supports booting from USB by default. If the system on eMMC has been formatted, you need to refer to **the method of burning a Linux image to eMMC** to burn the Linux image to eMMC.

2) Then you need to prepare a USB storage device, such as a USB flash drive.

3) Then please refer to the instructions in two sections: **the method of burning Linux images to TF cards based on Windows PC** and **the method of burning Linux images to TF cards based on Ubuntu PC** to burn Linux images to USB storage devices. There is no difference between burning a Linux image to a USB storage device and burning a Linux image to a TF card (when the TF card is inserted into the card reader, the card reader is actually equivalent to a USB flash drive).

4) Then insert the USB storage device that has burned the Linux system into the blue USB interface of the development board.



5) Then inserting the Type C power supply will start the linux system in the USB storage device.

6) After starting the system in the USB storage device, use the **df -h** command to see the actual capacity of the USB storage device.

```
orangepi@orangepi:~$ df -h
Filesystem          Size   Used Avail Use% Mounted on
tmpfs                784M   9.8M   774M     2% /run
/dev/sda1           15G    1.6G     13G    11% /
tmpfs                3.9G       0   3.9G     0% /dev/shm
tmpfs                5.0M       0    5.0M     0% /run/lock
tmpfs                3.9G       0   3.9G     0% /tmp
orangepi-ramlog      50M    1.4M     49M     3% /var/log
tmpfs                784M   4.0K   784M      1% /run/user/1001
```

2. 7. Launch the Orange Pie development board

1) The eMMC of the development board comes pre installed with the OpenWRT system, which can be used to test the full functionality of the development board.

2) The development board has an Ethernet port that can be plugged into a network cable for internet access.

3) Connect a **high-quality** power adapter with a 5V/3A USB Type-C interface.

Remember not to insert a power adapter with a voltage output greater than 5V, as it may burn out the development board.

Many unstable phenomena during the power on startup process of the system are basically caused by power supply problems, so a reliable power adapter is very important. If you notice continuous restarts during the startup process, please replace the power supply or Type-C data cable and try again.

The Type-C power interface does not support PD negotiation.

Also, please do not connect the USB port of the computer to power the development board.

4) Then turn on the power adapter switch.

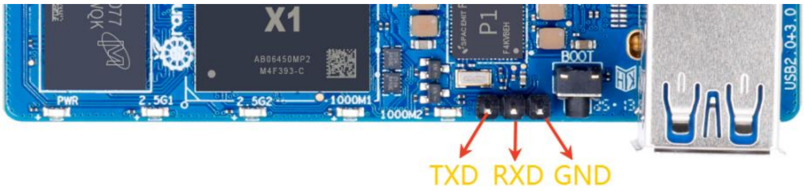
5) Please connect the development board to the computer using a serial port cable. For instructions on how to connect the serial port, please refer to the section on **debugging serial port usage**.

2. 8. Debug Serial Port Usage

2. 8. 1. Debug serial port connection instructions

1) First, you need to prepare a 3.3V USB to TTL module, and then insert the USB interface of the USB to TTL module into the USB interface of the computer.

2) The corresponding relationship between the debugging serial port GND, RXD and TXD pins of the development board is shown in the figure below:



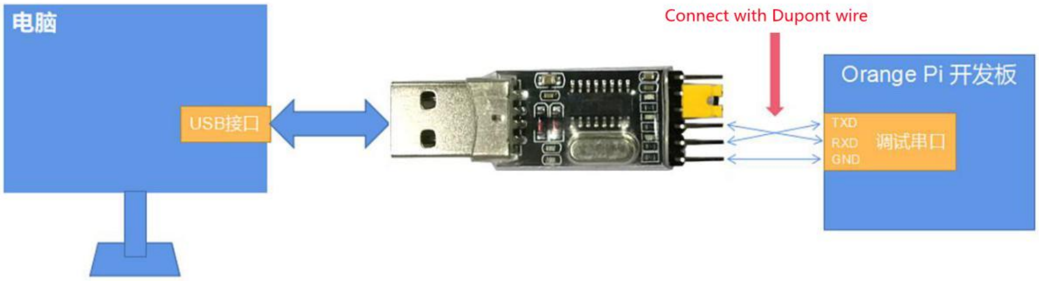
3) The GND, TXD and RXD pins of the USB to TTL module need to be connected to the debug serial port of the development board through DuPont cables.

a. Connect the GND of the USB to TTL module to the GND of the development board.

b. Connect the RX pin of the USB-to-TTL module to the TX pin of the development board.

c. Connect the TX pin of the USB-to-TTL module to the RX pin of the development board.

4) The connection diagram of the USB-to-TTL module between the computer and the Orange Pi development board is shown below:



Schematic diagram of connecting the USB to TTL module to the computer and the Orange Pi development board

For serial connections, TX and RX must be crossed. To simplify, try random connection first - if no output appears, just swap TX and RX. One configuration will always work.

2. 8. 2. How to use the debugging serial port on Ubuntu platform

There are many serial port debugging software that can be used under Linux, such as putty, minicom, etc. The following demonstrates how to use putty.

1) First, insert the USB-to-TTL module into your Ubuntu computer's USB port. If the module is properly detected, you should see its corresponding device node under /dev/ directory. Note down this device name as it will be needed when configuring the serial

terminal software later.

```
test@test:~$ ls /dev/ttyUSB*
/dev/ttyUSB0
```

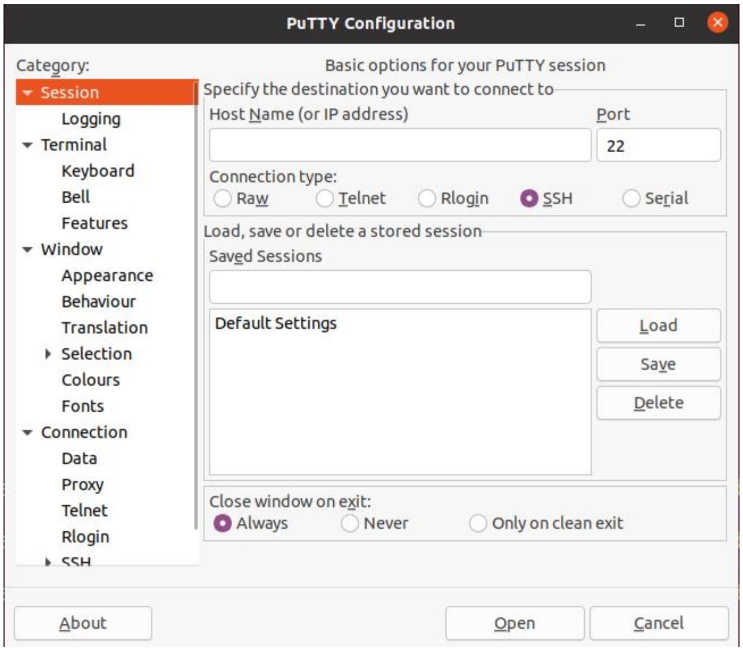
2) Then install putty on your Ubuntu PC using the command below.

```
test@test:~$ sudo apt-get update
test@test:~$ sudo apt-get install -y putty
```

3) Then run putty and remember to add sudo permissions.

```
test@test:~$ sudo putty
```

4) After executing the putty command, the following interface will pop up.



PuTTY Configuration

Basic options for your PuTTY session

Specify the destination you want to connect to

Host Name (or IP address)

Port: 22

Connection type: ○ Raw ○ Telnet ○ Rlogin ● SSH ○ Serial

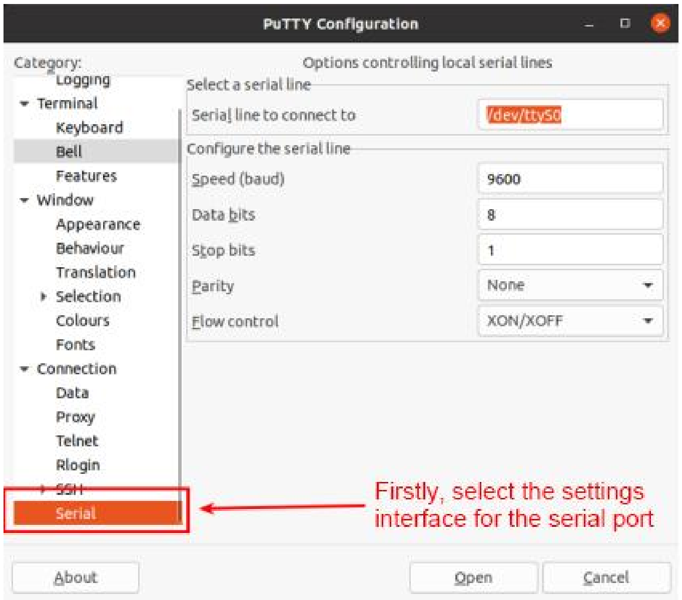
Load, save or delete a stored session

Saved Sessions

Default Settings

Close window on exit: ● Always ○ Never ○ Only on clean exit

5) First select the serial port settings interface.



Options controlling local serial lines

Select a serial line

Serial line to connect to: /dev/ttyS0

Configure the serial line

Speed (baud): 9600

Data bits: 8

Stop bits: 1

Parity: None

Flow control: XON/XOFF

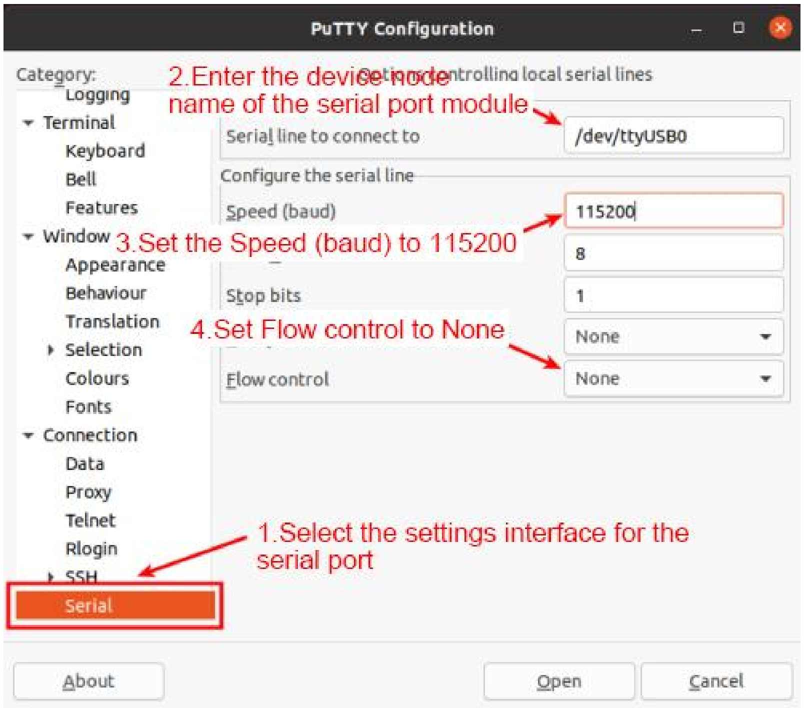
Firstly, select the settings interface for the serial port

6) Then set the parameters of the serial port.

a. Set **Serial line to connect to** to **/dev/ttyUSB0** (change to the corresponding node name, usually **/dev/ttyUSB0**).

b. Set **Speed(baud)** to 115200 (the baud rate of the serial port).

c. Set **Flow control** to None.



Options controlling local serial lines

Serial line to connect to: /dev/ttyUSB0

Configure the serial line

Speed (baud): 115200

8

Stop bits: 1

None

Flow control: None

2.Enter the device node name of the serial port module

3.Set the Speed (baud) to 115200

4.Set Flow control to None

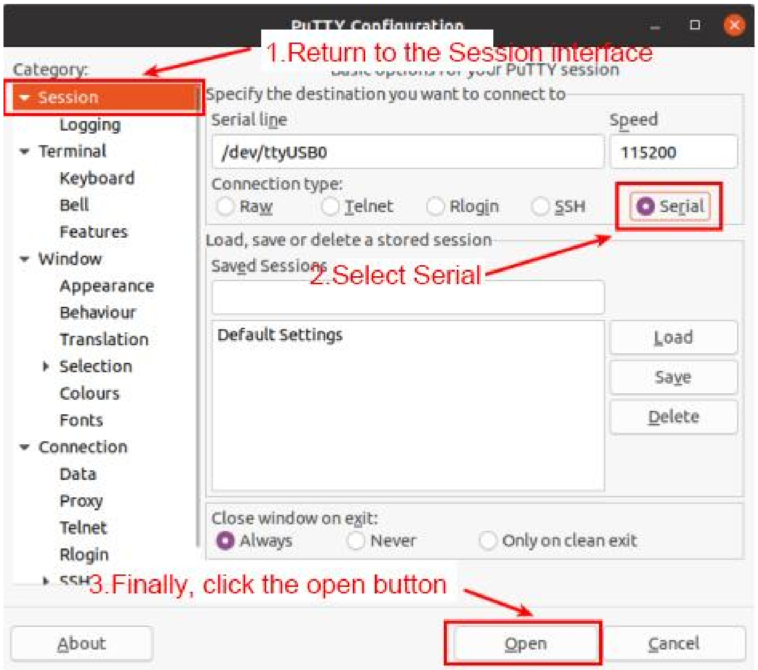
1.Select the settings interface for the serial port

7) After completing the settings on the serial port settings interface, return to the Session

interface.

a. First select **Connection type** as Serial.

b. Then click the **Open** button to connect the serial port.



1.Return to the Session interface

2.Select Serial

3.Finally, click the open button

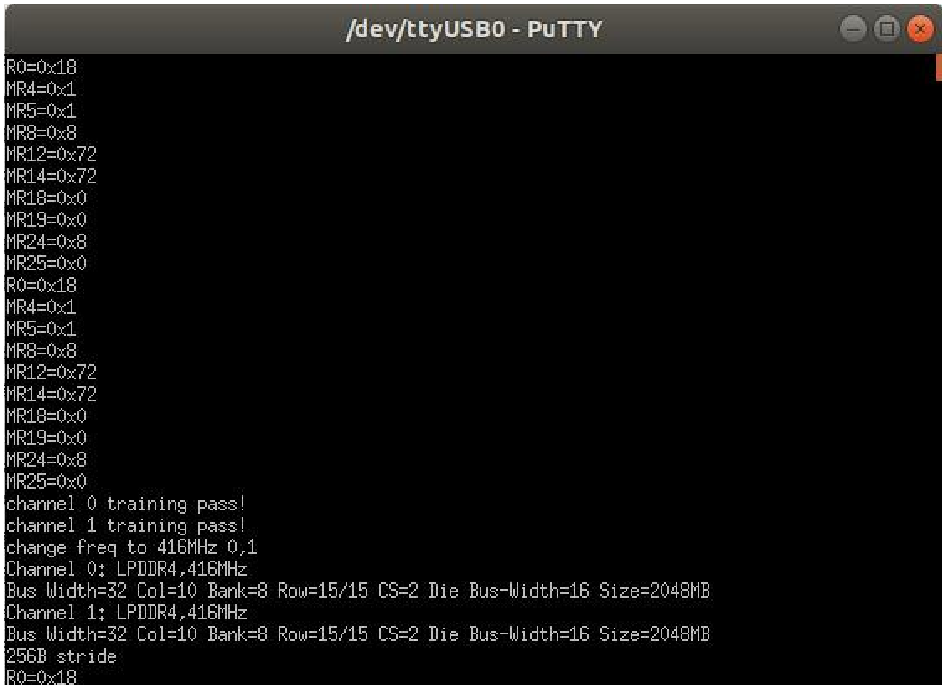
Serial line: /dev/ttyUSB0

Speed: 115200

Connection type: ○ Raw ○ Telnet ○ Rlogin ○ SSH ● Serial

Close window on exit: ● Always ○ Never ○ Only on clean exit

8) After starting the development board, you can see the log information output by the system from the opened serial port terminal.



/dev/ttyUSB0 - PuTTY

```
R0=0x18
MR4=0x1
MR5=0x1
MR8=0x8
MR12=0x72
MR14=0x72
MR18=0x0
MR19=0x0
MR24=0x8
MR25=0x0
R0=0x18
MR4=0x1
MR5=0x1
MR8=0x8
MR12=0x72
MR14=0x72
MR18=0x0
MR19=0x0
MR24=0x8
MR25=0x0
channel 0 training pass!
channel 1 training pass!
change freq to 416MHz 0,1
Channel 0: LPDDR4,416MHz
Bus Width=32 Col=10 Bank=8 Row=15/15 CS=2 Die Bus-Width=16 Size=2048MB
Channel 1: LPDDR4,416MHz
Bus Width=32 Col=10 Bank=8 Row=15/15 CS=2 Die Bus-Width=16 Size=2048MB
256B stride
R0=0x18
```

2. 8. 3. How to use the debug serial port on Windows

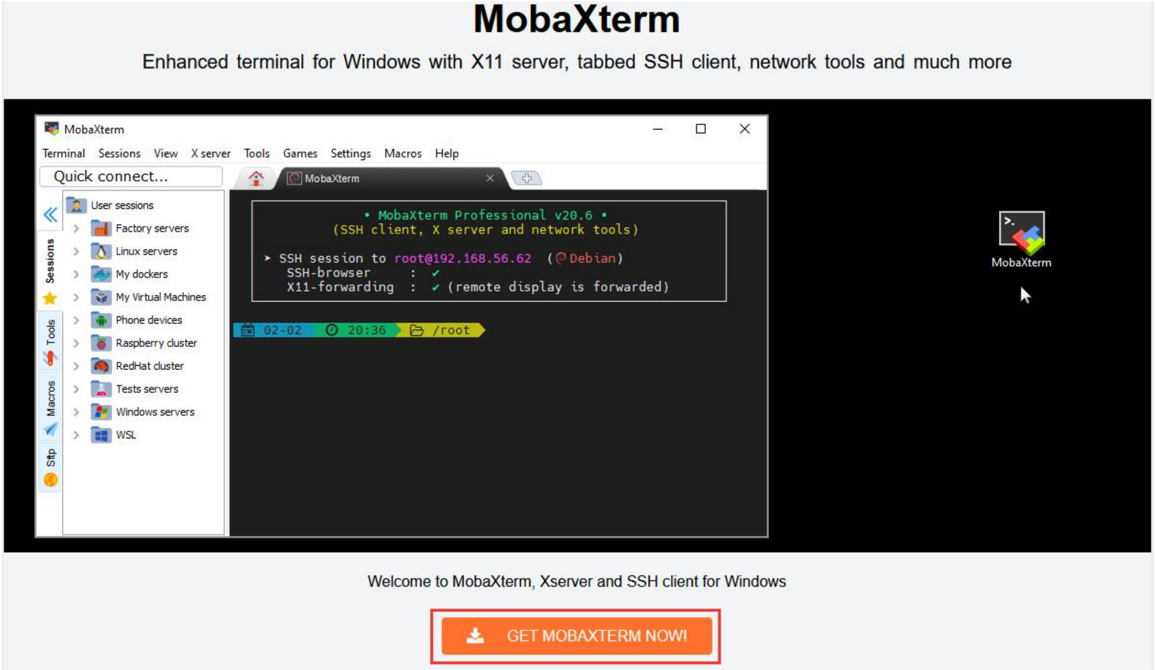
There are many serial port debugging software that can be used under Windows, such as SecureCRT, MobaXterm, etc. The following demonstrates how to use MobaXterm. This software has a free version and can be used without purchasing a serial number.

1) Download MobaXterm.

a. Download MobaXterm from the following URL:

https://mobaxterm.mobatek.net

b. Go to the MobaXterm download page and click **GET XOBATERM NOW!**.



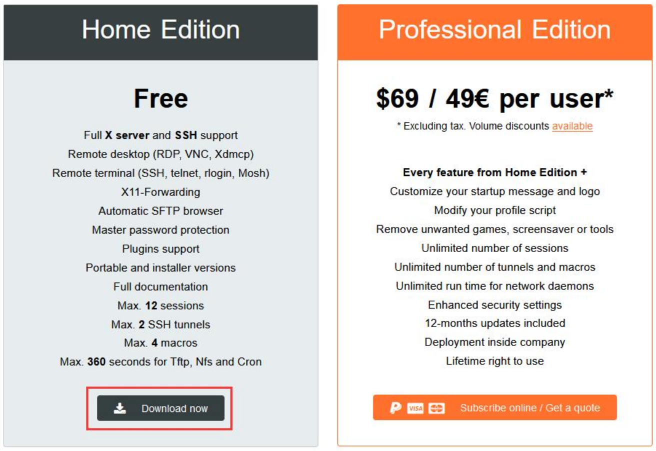
MobaXterm

Enhanced terminal for Windows with X11 server, tabbed SSH client, network tools and much more

```
      • MobaXterm Professional v20.6 •
   (SSH client, X server and network tools)

 ➤ SSH session to root@192.168.56.62  (Debian)
   SSH-browser     :  ✔
   X11-forwarding  :  ✔ (remote display is forwarded)

 02-02   20:36   /root
```

Welcome to MobaXterm, Xserver and SSH client for Windows

c. Then choose to download the Home Edition.



Home Edition

Free

Full **X server** and **SSH** support
Remote desktop (RDP, VNC, Xdmcp)
Remote terminal (SSH, telnet, rlogin, Mosh)
X11-Forwarding
Automatic SFTP browser
Master password protection
Plugins support
Portable and installer versions
Full documentation
Max. **12** sessions
Max. **2** SSH tunnels
Max. **4** macros
Max. **360** seconds for Tftp, Nfs and Cron

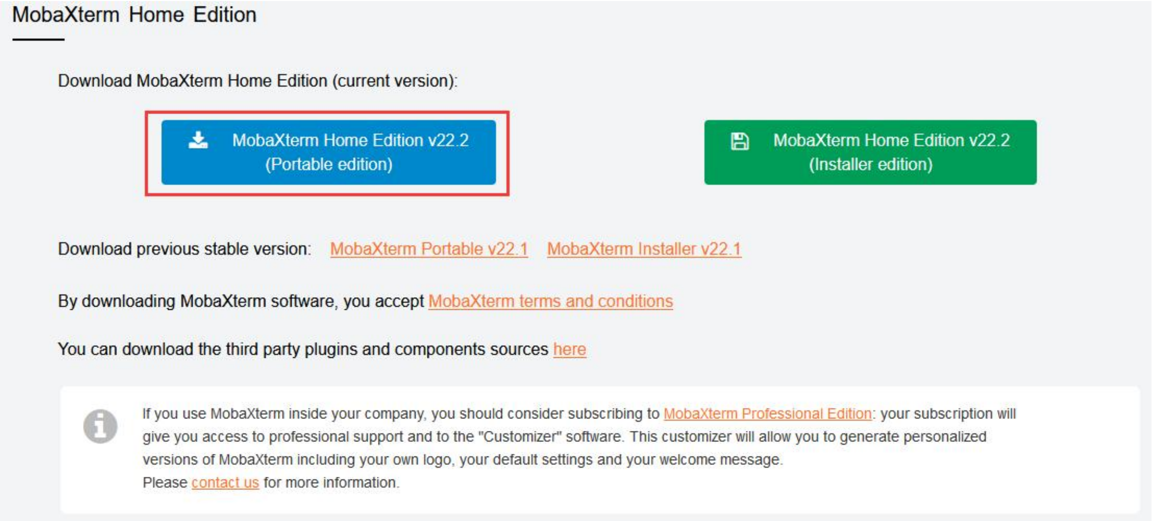
Professional Edition

$69 / 49€ per user*

* Excluding tax. Volume discounts available

Every feature from Home Edition +
Customize your startup message and logo
Modify your profile script
Remove unwanted games, screensaver or tools
Unlimited number of sessions
Unlimited number of tunnels and macros
Unlimited run time for network daemons
Enhanced security settings
12-months updates included
Deployment inside company
Lifetime right to use

d. Then select the Portable version. After downloading, you don’t need to install it. You can use it directly by opening it.



MobaXterm Home Edition

Download MobaXterm Home Edition (current version):

MobaXterm Home Edition v22.2 (Portable edition)
MobaXterm Home Edition v22.2 (Installer edition)

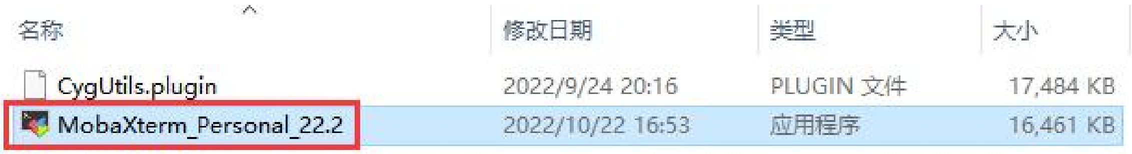
Download previous stable version: MobaXterm Portable v22.1 MobaXterm Installer v22.1

By downloading MobaXterm software, you accept MobaXterm terms and conditions

You can download the third party plugins and components sources here

If you use MobaXterm inside your company, you should consider subscribing to MobaXterm Professional Edition: your subscription will give you access to professional support and to the "Customizer" software. This customizer will allow you to generate personalized versions of MobaXterm including your own logo, your default settings and your welcome message.
Please contact us for more information.

2) After downloading, use decompression software to decompress the downloaded compressed package to get the executable software of MobaXterm, and then double-click to open it.



名称	修改日期	类型	大小
CygUtils.plugin	2022/9/24 20:16	PLUGIN 文件	17,484 KB
MobaXterm_Personal_22.2	2022/10/22 16:53	应用程序	16,461 KB

3) After opening the software, the steps to set up the serial port connection are as follows:

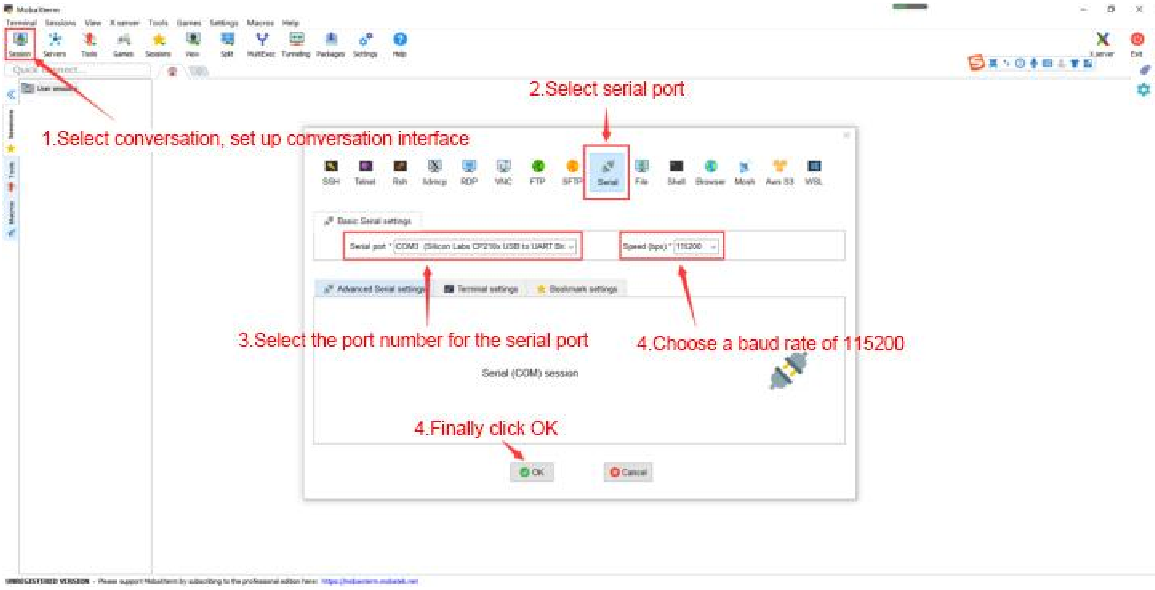
a. Open the session settings interface.

b. Select the serial port type.

c. Select the serial port number (select the corresponding port number according to the actual situation). If you cannot see the port number, use Driver Master to scan and install the USB to TTL serial port chip driver.

d. Select the serial port baud rate as 115200.

e. Finally, click the "OK" button to complete the settings.


MobaXterm
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
X server Exit
Quick connect...
User sessions
SSH Telnet Rsh Xdmcp RDP VNC FTP SFTP Serial File Shell Browser Mosh Aws S3 WSL
UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

1.Select conversation, set up conversation interface

2.Select serial port

Basic Serial settings

Serial port * COM3 (Silicon Labs CP210x USB to UART Br

Speed (bps) * 115200

Advanced Serial settings | Terminal settings | Bookmark settings

3.Select the port number for the serial port

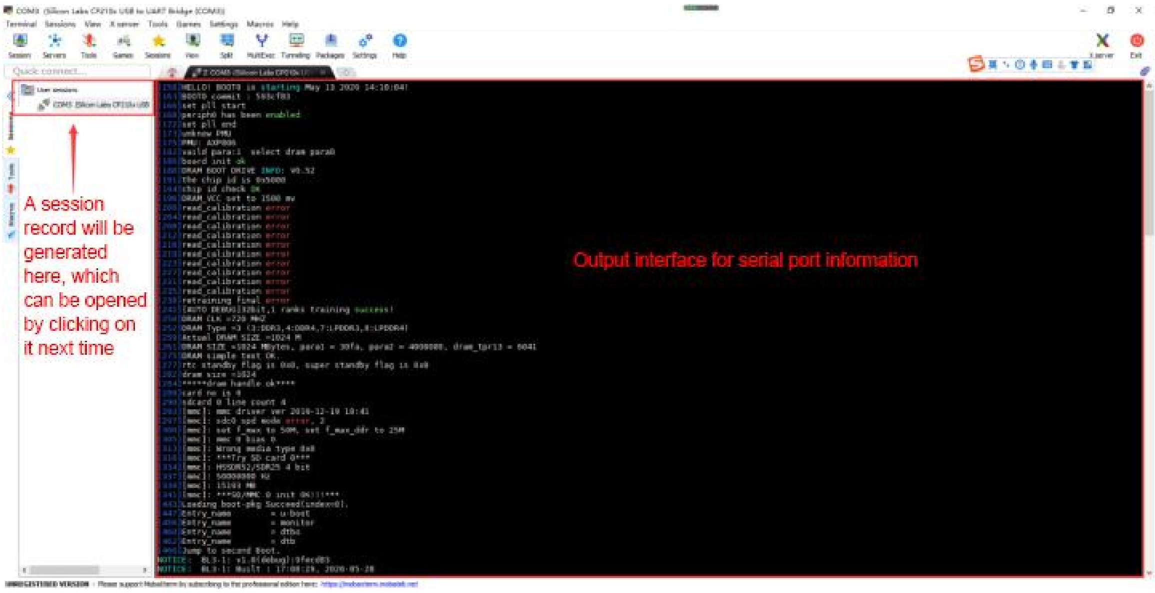
4.Choose a baud rate of 115200

Serial (COM) session

4.Finally click OK

OK Cancel

4) Click the "**OK**" button to enter the following interface. Now start the development board and you can see the output information of the serial port.


COM3 (Silicon Labs CP210x USB to UART Bridge (COM3))
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
X server Exit
Quick connect...
2. COM3 (Silicon Labs CP210x U
User sessions
COM3 (Silicon Labs CP210x USB
UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

```
[158]HELLO! BOOT0 is starting May 13 2020 14:10:04!
[163]BOOT0 commit : 593cf83
[166]set pll start
[168]periph0 has been enabled
[172]set pll end
[173]unknow PMU
[175]PMU: AXP806
[182]vaild para:1  select dram para0
[188]board init ok
[188]DRAM BOOT DRIVE INFO: V0.52
[191]the chip id is 0x5000
[194]chip id check OK
[196]DRAM_VCC set to 1500 mv
[200]read_calibration error
[204]read_calibration error
[208]read_calibration error
[212]read_calibration error
[216]read_calibration error
[219]read_calibration error
[223]read_calibration error
[227]read_calibration error
[231]read_calibration error
[235]read_calibration error
[238]retraining final error
[242][AUTO DEBUG]32bit,1 ranks training success!
[248]DRAM CLK =720 MHZ
[252]DRAM Type =3 (3:DDR3,4:DDR4,7:LPDDR3,8:LPDDR4)
[255]Actual DRAM SIZE =1024 M
[261]DRAM SIZE =1024 MBytes, para1 = 30fa, para2 = 4000000, dram_tpr13 = 6041
[275]DRAM simple test OK.
[277]rtc standby flag is 0x0, super standby flag is 0x0
[282]dram size =1024
[284]*****dram handle ok****
[288]card no is 0
[290]sdcard 0 line count 4
[293][mmc]: mmc driver ver 2018-12-19 10:41
[297][mmc]: sdc0 spd mode error, 2
[300][mmc]: set f_max to 50M, set f_max_ddr to 25M
[305][mmc]: mmc 0 bias 0
[313][mmc]: Wrong media type 0x0
[316][mmc]: ***Try SD card 0***
[334][mmc]: HSSDR52/SDR25 4 bit
[337][mmc]: 50000000 Hz
[340][mmc]: 15193 MB
[341][mmc]: ***SD/MMC 0 init OK!!!***
[443]Loading boot-pkg Succeed(index=0).
[447]Entry_name        = u-boot
[456]Entry_name        = monitor
[460]Entry_name        = dtbo
[462]Entry_name        = dtb
[466]Jump to second Boot.
NOTICE:  BL3-1: v1.0(debug):9fecd83
NOTICE:  BL3-1: Built : 17:08:28, 2020-05-28
```

A session record will be generated here, which can be opened by clicking on it next time

Output interface for serial port information

3. Ubuntu Server system usage instructions

3. 1. Supported Linux image types and kernel versions

Linux Image Type	Kernel version	Server version
Ubuntu 24.04 - Noble	Linux6.6	Support

3. 2. Linux 6.6 system compatibility

Function	Ubuntu24.04
USB2.0x1	OK
USB3.0x1	OK
USB boot system	OK
3pin debug serial port	OK
eMMC Start	OK
Gigabit Ethernet port x2	OK
2.5G Ethernet port x2	OK
Ethernet port status light	OK
Watchdog test	OK

3. 3. Linux command format description in this manual

1) All commands in this manual that need to be entered in the Linux system will be framed with the following boxes.

As shown below, the contents in the yellow box indicate the contents that require special attention, except for the commands inside.

2) Description of the prompt type before the command.

a. The prompt before the command refers to the content in the red part of the box below. This part is not part of the Linux command, so when entering a command in the Linux system, please do not enter the content in red font.

```
orangepi@orangepi:~$ sudo apt update
root@orangepi:~# vim /boot/boot.cmd
test@test:~$ ssh root@192.168.1.xxx
root@test:~# ls
```

b. **root@orangepi:~$** The prompt indicates that this command is entered in the Linux system of the development board. The**$** at the end of the prompt indicates that the current user of the system is a normal user. When executing privileged commands, you need to add **sudo.**

c. **root@orangepi:~#** The prompt indicates that this command is entered in the Linux system of the development board. The # at the end of the prompt indicates that the current user of the system is the root user and can execute any command he wants.

d. **test@test:~$** The prompt indicates that this command is entered in an Ubuntu PC or Ubuntu virtual machine, not in the Linux system of the development board. The **$** at the end of the prompt indicates that the current user of the system is a normal user. When executing privileged commands, **sudo** needs to be added.

e. **root@test:~#** The prompt indicates that this command is entered in an Ubuntu PC or Ubuntu virtual machine, not in the Linux system of the development board. The # at the end of the prompt indicates that the current user of the system is the root user and can execute any command you want.

3) What are the commands that need to be entered?

a. As shown below, the bold black part is the command that needs to be entered, and the content below the command is the output (some commands have output, some may not). This part does not need to be entered.

```
root@orangepi:~# cat /boot/orangepiEnv.txt
verbosity=7
bootlogo=false
console=serial
```

b. As shown below, some commands cannot fit in one line and will be placed on the next line. The bold black parts are the commands that need to be entered. When these commands are entered on one line, the "\" at the end of each line needs to be removed, as it is not part of the command. In addition, there are spaces between different parts of the command, so please do not miss them.

```
orangepi@orangepi:~$ echo    \
"deb [arch=$(dpkg --print-architecture)    \
signed-by=/usr/share/keyrings/docker-archive-keyring.gpg]    \
https://download.docker.com/linux/debian    \
$(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

3. 4. Linux system login instructions

3. 4. 1. Linux system default login account and password

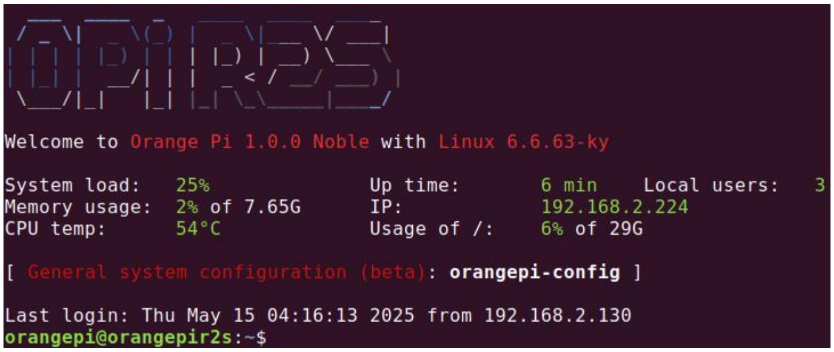
Account	Password
root	orangepi
orangepi	orangepi

Please note that when you enter the password, the specific content of the password will not be displayed on the screen. Please do not think that there is any malfunction. Just press Enter after entering it.

If you get an error message when entering the password, or there is a problem with the ssh connection, please note that as long as you are using the Linux image provided by Orange Pi, do not doubt that the password above is incorrect, but look for other reasons.

3. 4. 2. How to set up automatic login for Linux system terminal

1) The Linux system automatically logs in to the terminal by default, and the default login username is **orangepi**.



```
Welcome to Orange Pi 1.0.0 Noble with Linux 6.6.63-ky

System load:   25%               Up time:       6 min     Local users:   3
Memory usage:  2% of 7.65G       IP:            192.168.2.224
CPU temp:      54°C              Usage of /:    6% of 29G

[ General system configuration (beta): orangepi-config ]

Last login: Thu May 15 04:16:13 2025 from 192.168.2.130
orangepi@orangepir2s:~$
```

2) Use the following command to set the root user to automatically log in to the terminal.

```
orangepi@orangepi:~$ sudo auto_login_cli.sh root
```

3) Use the following command to disable automatic login to the terminal.

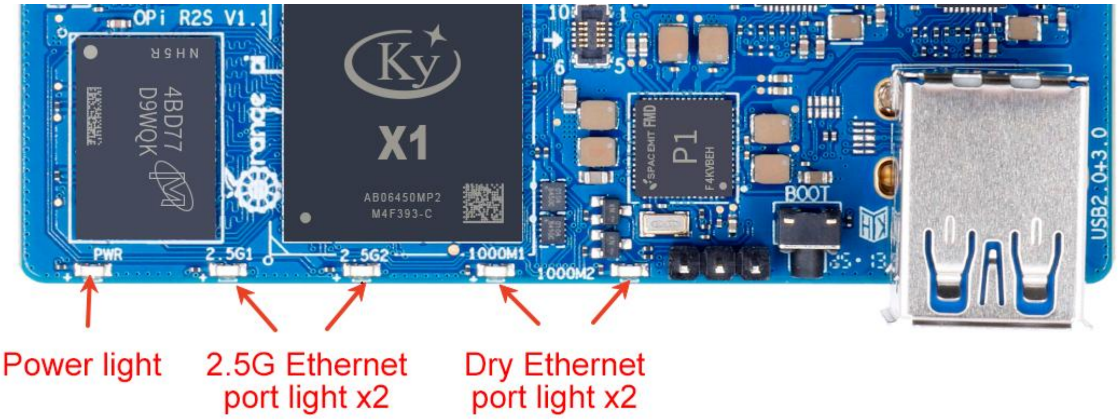
```
orangepi@orangepi:~$ sudo auto_login_cli.sh -d
```

4) Use the following command to set the orangepi user to automatically log in to the terminal again.

```
orangepi@orangepi:~$ sudo auto_login_cli.sh orangepi
```

3. 5. Onboard LED Light Description

1) There are five LED lights on the development board, one power light, two 2.5G network port lights, and two Gigabit network port lights. The locations are shown in the figure below:



2) As long as the development board is powered on, the power light will be on. This is controlled by hardware and cannot be turned off by software. The red LED light can be used to determine whether the power of the development board has been turned on normally.

3) The network port light will flash when data transmission occurs at the corresponding network port, which is also controlled by the network port chip itself.

3. 6. Network connection test

3. 6. 1. Ethernet port test

1) The development board has two Gigabit Ethernet interfaces and two 2.5G Ethernet interfaces. The test methods for these four network interfaces are the same. First, insert one end of the network cable into the Ethernet interface of the development board, and connect the other end of the network cable to the router, and make sure the network is unobstructed.

2) After the system starts, the IP address will be automatically assigned to the Ethernet card through DHCP, and no other configuration is required.

3) The command to check the IP address in the Linux system of the development board is as follows:

```
orangepi@orangepi:~$ ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: end0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:e0:4c:68:00:13 brd ff:ff:ff:ff:ff:ff
    inet 192.168.2.241/24 brd 192.168.2.255 scope global dynamic noprefixroute end0
       valid_lft 43186sec preferred_lft 43186sec
    inet6 fdcd:e671:36f4::47c/128 scope global dynamic noprefixroute
       valid_lft 43186sec preferred_lft 43186sec
    inet6 fdcd:e671:36f4:0:5689:f699:84ec:d4cb/64 scope global temporary dynamic
       valid_lft 604786sec preferred_lft 85811sec
    inet6 fdcd:e671:36f4:0:52ab:6ce7:cfc7:9ecf/64 scope global mngtmpaddr noprefixroute
       valid_lft forever preferred_lft forever
    inet6 fe80::f082:90bd:3fbd:dc01/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
3: end1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:e0:4c:68:00:14 brd ff:ff:ff:ff:ff:ff
    inet 192.168.2.242/24 brd 192.168.2.255 scope global dynamic noprefixroute end1
       valid_lft 43179sec preferred_lft 43179sec
    inet6 fdcd:e671:36f4::49e/128 scope global dynamic noprefixroute
       valid_lft 43177sec preferred_lft 43177sec
    inet6 fdcd:e671:36f4:0:da95:4c2f:806f:5617/64 scope global temporary dynamic
       valid_lft 604777sec preferred_lft 85899sec
    inet6 fdcd:e671:36f4:0:7d9:7510:ccc5:fac9/64 scope global mngtmpaddr noprefixroute
       valid_lft forever preferred_lft forever
    inet6 fe80::db62:da89:a277:2ff0/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
4: enP1p1s0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc mq state DOWN group default qlen 1000
    link/ether 00:e0:4c:68:00:01 brd ff:ff:ff:ff:ff:ff
5: enP2p1s0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc mq state DOWN group default qlen 1000
    link/ether 00:e0:4c:68:00:02 brd ff:ff:ff:ff:ff:ff
```

When using ifconfig to check the IP address, if the following message is displayed, it is because sudo is not added. The correct command is: **sudo ifconfig**.

```
orangepi@orangepi:~$ ifconfig
Command 'ifconfig' is available in the following places
 * /sbin/ifconfig
 * /usr/sbin/ifconfig
The command could not be located because '/sbin:/usr/sbin' is not included in the PATH environment variable.
This is most likely caused by the lack of administrative privileges associated with your user account.
ifconfig: command not found
```

There are two ways to check the IP address after the development board is started:

1. Enter the **ip addr show** command in the debugging serial port terminal to view the IP address.

2. If there is no debugging serial port, you can also view the IP address of the

development board's network port through the router's management interface. However, this method often fails to properly see the IP address of the development board. If you cannot see it, the debugging method is as follows:

A) Check if the network cable is plugged in tightly, or try another one;

B) Try another router (I have encountered many router problems, such as the router cannot allocate IP addresses normally, or the IP addresses are allocated normally but cannot be seen in the router);

C) If there is no router to replace, you can only use the debug serial port to view the IP address.

It should also be noted that the development board DHCP automatically assigns IP addresses without any settings.

4) The command to test network connectivity is as follows. The **ping** command can be interrupted by pressing the **Ctrl+C**shortcut key.

```
orangepi@orangepi:~$ ping www.baidu.com -I end0         #Test command for one of the network ports
PING www.a.shifen.com (183.2.172.42) from 192.168.2.241 end0: 56(84) bytes of data.
64 bytes from 183.2.172.42: icmp_seq=1 ttl=53 time=10.1 ms
64 bytes from 183.2.172.42: icmp_seq=2 ttl=53 time=10.0 ms
64 bytes from 183.2.172.42: icmp_seq=3 ttl=53 time=9.91 ms
^C
--- www.a.shifen.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 9.910/10.017/10.126/0.088 ms
```

3. 6. 2. How to set a static IP address

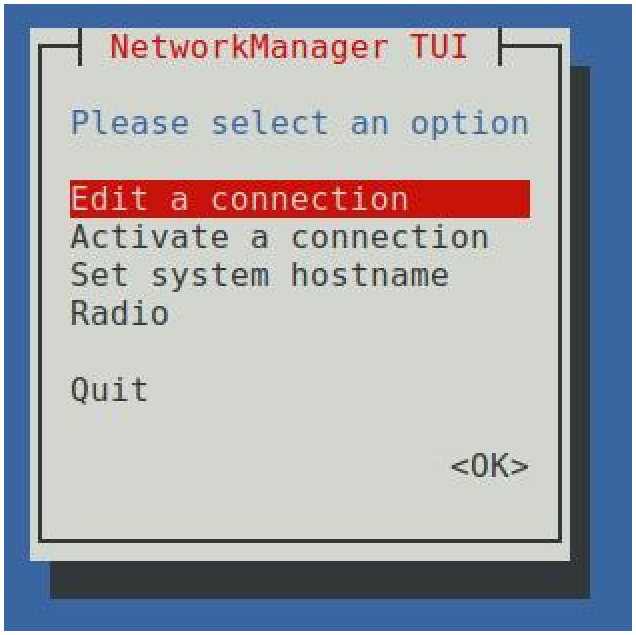
Please do not set a static IP address by modifying the /etc/network/interfaces configuration file.

3. 6. 2. 1. Using nmtui command to set static IP address

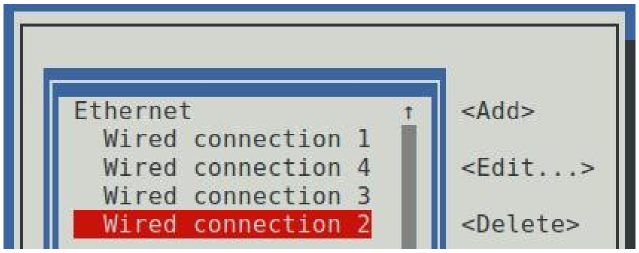
1) First run the **nmtui** command.

```
orangepi@orangepi:~$ sudo nmtui
```

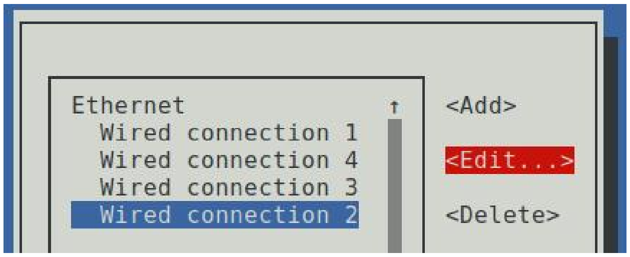
2) Then select **Edit a connection** and press Enter.



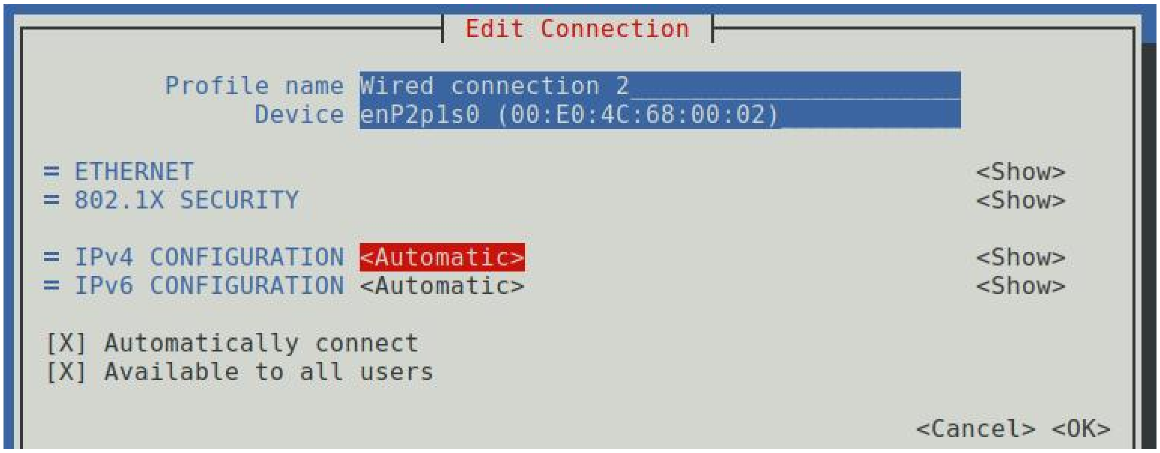
3) Then select the network interface for which you want to set a static IP address. For example, to set a static IP address for an **Ethernet** interface, select one of the interfaces in the figure below.



4) Then select **Edit**using the **Tab** key and press Enter.

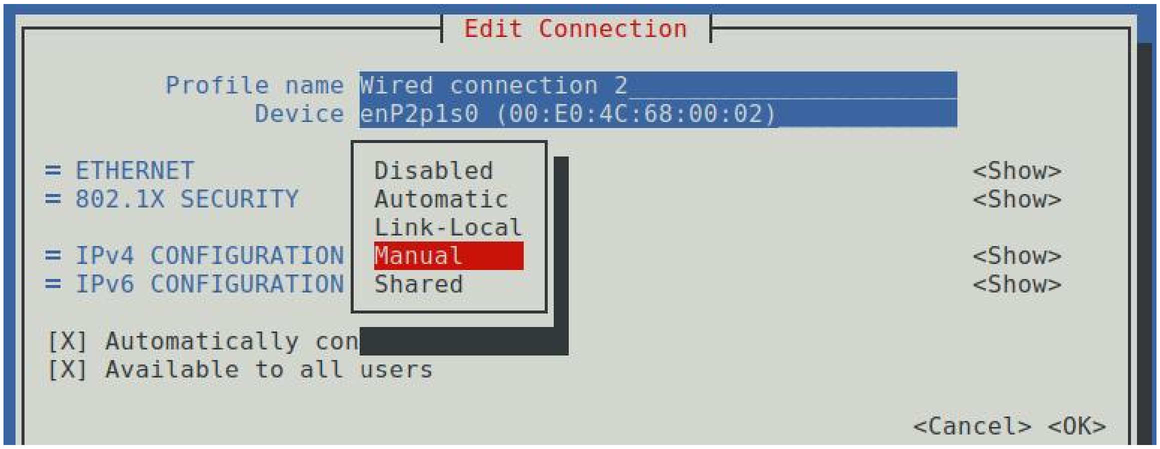


5) Then use the Tab key to move the cursor to the **<Automatic>** position shown in the figure below to configure IPv4.



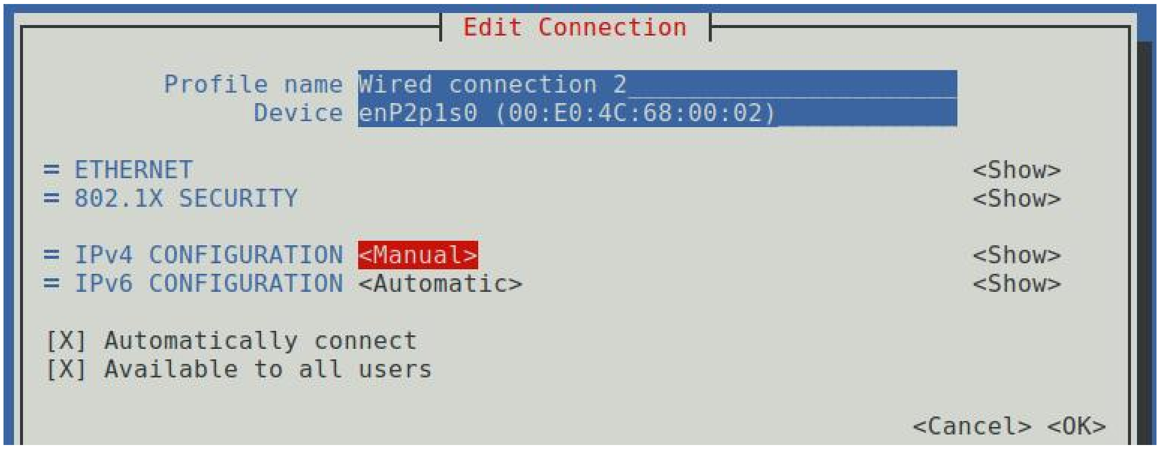
```
                         Edit Connection

       Profile name Wired connection 2
             Device enP2p1s0 (00:E0:4C:68:00:02)

 = ETHERNET                                                  <Show>
 = 802.1X SECURITY                                           <Show>

 = IPv4 CONFIGURATION <Automatic>                            <Show>
 = IPv6 CONFIGURATION <Automatic>                            <Show>

 [X] Automatically connect
 [X] Available to all users

                                                       <Cancel> <OK>
```

6) Press Enter, use the up and down arrow keys to select **Manual**, and then press Enter to confirm.



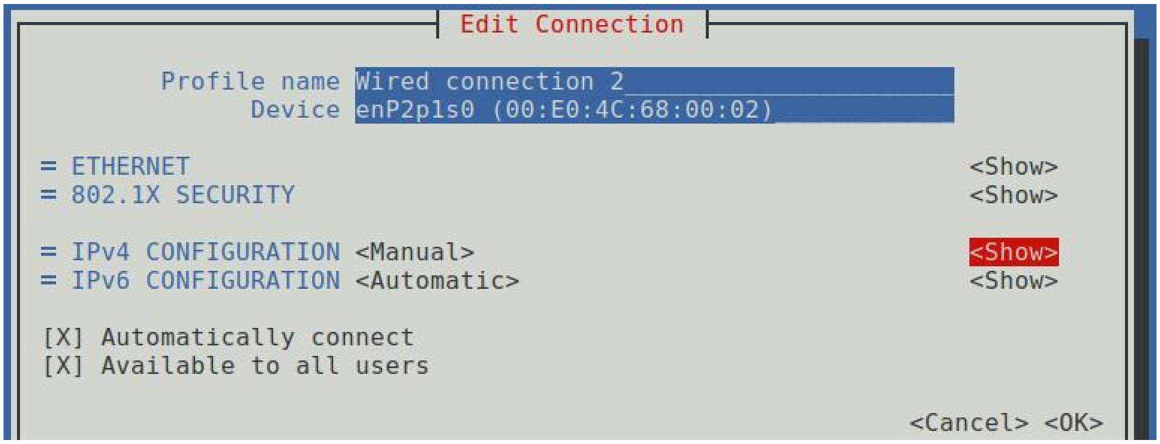
```
                         Edit Connection

       Profile name Wired connection 2
             Device enP2p1s0 (00:E0:4C:68:00:02)

 = ETHERNET            Disabled                              <Show>
 = 802.1X SECURITY     Automatic                             <Show>
                       Link-Local
 = IPv4 CONFIGURATION  Manual                                <Show>
 = IPv6 CONFIGURATION  Shared                                <Show>

 [X] Automatically con
 [X] Available to all users

                                                       <Cancel> <OK>
```

7) After selection, the display is as shown below:



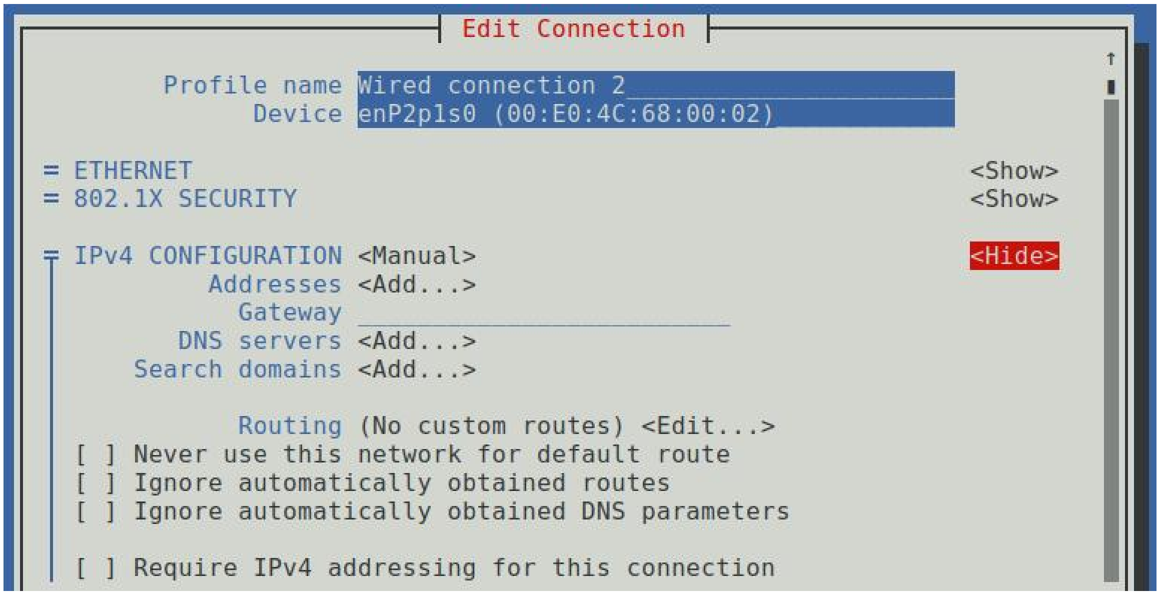
```
                         Edit Connection

       Profile name Wired connection 2
             Device enP2p1s0 (00:E0:4C:68:00:02)

 = ETHERNET                                                  <Show>
 = 802.1X SECURITY                                           <Show>

 = IPv4 CONFIGURATION <Manual>                               <Show>
 = IPv6 CONFIGURATION <Automatic>                            <Show>

 [X] Automatically connect
 [X] Available to all users

                                                       <Cancel> <OK>
```

8) Then use the Tab key to move the cursor to **<Show>**.



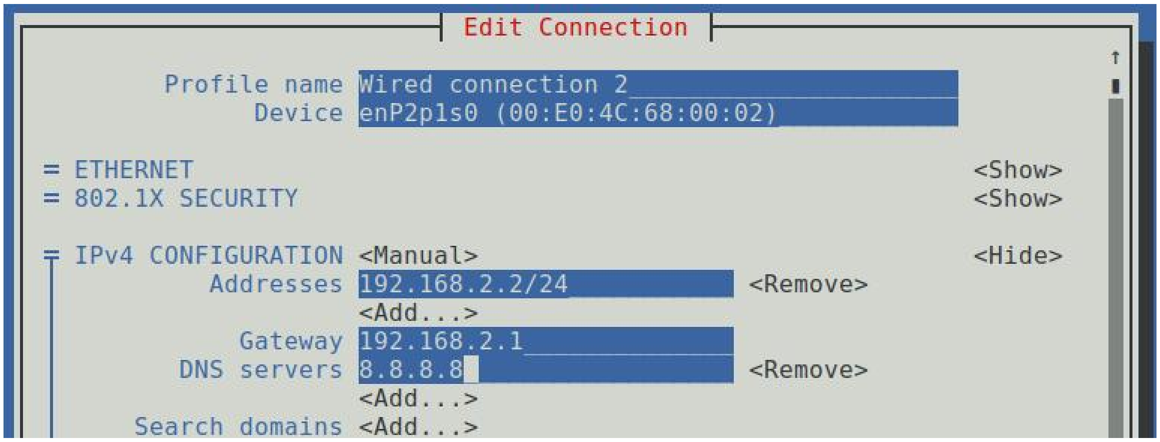
```
                    ┤ Edit Connection ├

        Profile name Wired connection 2
              Device enP2p1s0 (00:E0:4C:68:00:02)

= ETHERNET                                             <Show>
= 802.1X SECURITY                                      <Show>

= IPv4 CONFIGURATION <Manual>                          <Show>
= IPv6 CONFIGURATION <Automatic>                       <Show>

[X] Automatically connect
[X] Available to all users

                                                  <Cancel> <OK>
```

9) Then press Enter, and the following setting interface will pop up.



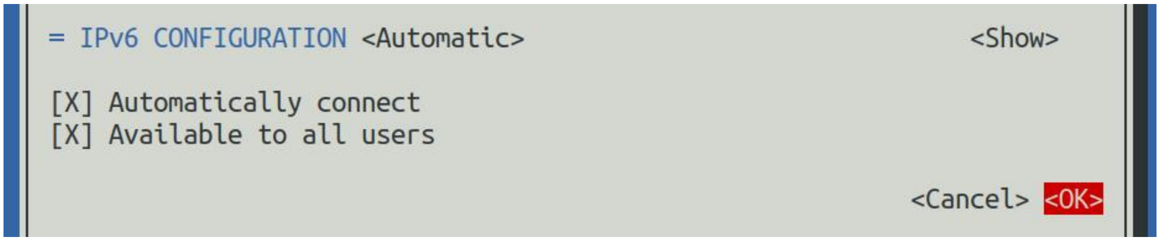
```
                    ┤ Edit Connection ├

        Profile name Wired connection 2
              Device enP2p1s0 (00:E0:4C:68:00:02)

= ETHERNET                                             <Show>
= 802.1X SECURITY                                      <Show>

╤ IPv4 CONFIGURATION <Manual>                          <Hide>
│          Addresses <Add...>
│            Gateway ________________________
│        DNS servers <Add...>
│     Search domains <Add...>
│
│            Routing (No custom routes) <Edit...>
│ [ ] Never use this network for default route
│ [ ] Ignore automatically obtained routes
│ [ ] Ignore automatically obtained DNS parameters
│
│ [ ] Require IPv4 addressing for this connection
```

10) Then you can set the IP address (Addresses), gateway (Gateway) and DNS server address as shown in the figure below (there are many other setting options, please explore them yourself). Please set them according to your specific needs. The value set in the figure below is just an example.



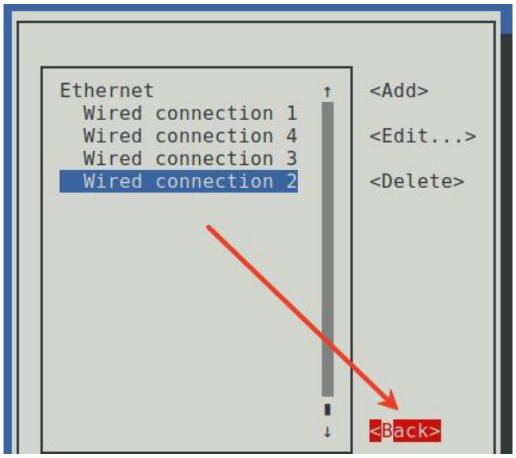
```
                         Edit Connection

      Profile name Wired connection 2
            Device enP2p1s0 (00:E0:4C:68:00:02)

= ETHERNET                                               <Show>
= 802.1X SECURITY                                        <Show>

╤ IPv4 CONFIGURATION <Manual>                            <Hide>
│          Addresses 192.168.2.2/24         <Remove>
│                    <Add...>
│            Gateway 192.168.2.1
│        DNS servers 8.8.8.8                <Remove>
│                    <Add...>
│     Search domains <Add...>
```

11) After setting, move the cursor to **<OK>** in the lower right corner and press Enter to confirm.



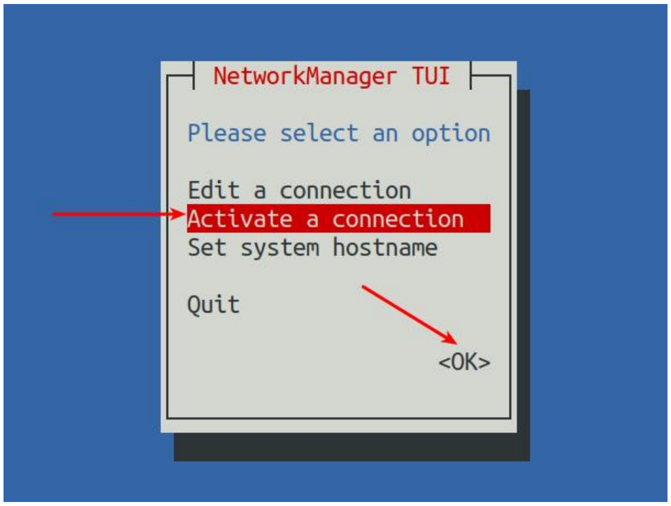
```
= IPv6 CONFIGURATION <Automatic>                        <Show>

[X] Automatically connect
[X] Available to all users

                                                 <Cancel> <OK>
```

12) Then click **<Back>** to return to the previous selection interface.



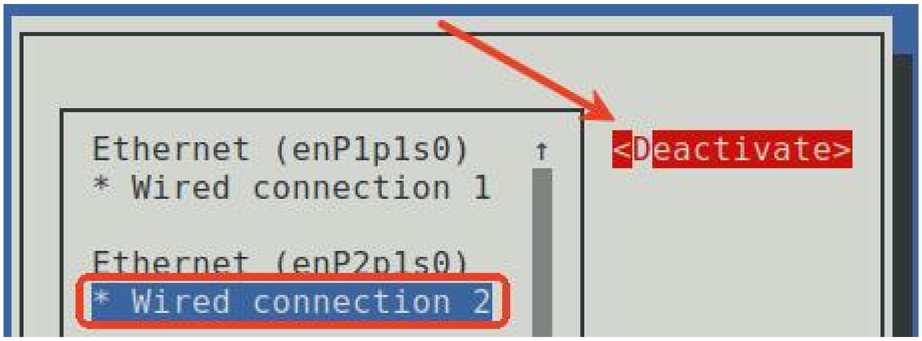
```
Ethernet                  <Add>
  Wired connection 1
  Wired connection 4      <Edit...>
  Wired connection 3
  Wired connection 2      <Delete>



                          <Back>
```

13) Then select **Activate a connection**, move the cursor to **<OK>**, and press Enter.



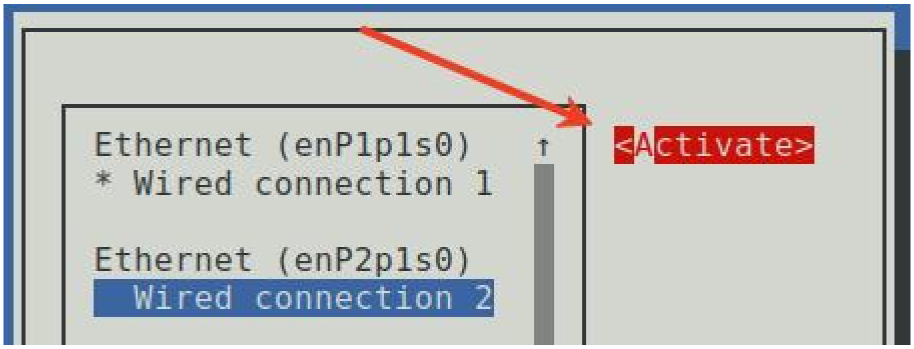
```
NetworkManager TUI

Please select an option

Edit a connection
Activate a connection
Set system hostname

Quit

                    <OK>
```

14) Then select the network interface you want to configure, such as **Wired connection 2**, move the cursor to **<Deactivate>**, and press Enter to disable **Wired connection 2**.



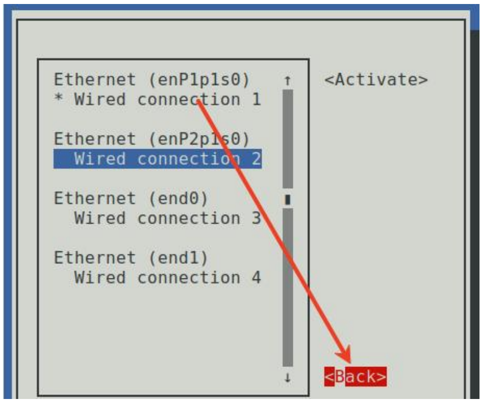
```
Ethernet (enP1p1s0)        ↑   <Deactivate>
* Wired connection 1

Ethernet (enP2p1s0)
* Wired connection 2
```

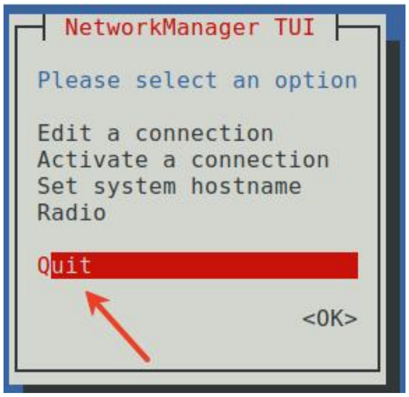
15) Then please do not move the cursor and press the Enter key to re-enable **Wired connection 2**, so that the static IP address set previously will take effect.



```
Ethernet (enP1p1s0)        ↑   <Activate>
* Wired connection 1

Ethernet (enP2p1s0)
  Wired connection 2
```

16) Then you can exit **nmtui** using the**<Back>** and **Quit** buttons.





17) Then use **ip addr show enP2p1s0** to see that the IP address of the network port has become the static IP address set previously.

```
orangepi@orangepi:~$ ip addr show enP2p1s0
3: enP2p1s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:e0:4c:68:00:14 brd ff:ff:ff:ff:ff:ff
    inet 192.168.2.2/24 brd 192.168.2.255 scope global noprefixroute end1
       valid_lft forever preferred_lft forever
    inet6 fdcd:e671:36f4::49e/128 scope global dynamic noprefixroute
       valid_lft 42950sec preferred_lft 42950sec
    inet6 fdcd:e671:36f4:0:2139:e484:d595:deda/64 scope global temporary dynamic
       valid_lft 604550sec preferred_lft 85735sec
    inet6 fdcd:e671:36f4:0:7d9:7510:ccc5:fac9/64 scope global mngtmpaddr noprefixroute
       valid_lft forever preferred_lft forever
    inet6 fe80::db62:da89:a277:2ff0/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

18) Then you can test the network connectivity to check whether the IP address is configured OK. The **ping** command can be interrupted by pressing the **Ctrl+C** shortcut key.

```
orangepi@orangepi:~$ ping www.baidu.com -I enP2p1s0
PING www.a.shifen.com (183.2.172.42) from 192.168.2.2 enP2p1s0: 56(84) bytes of data.
64 bytes from 183.2.172.42: icmp_seq=1 ttl=53 time=10.2 ms
64 bytes from 183.2.172.42: icmp_seq=2 ttl=53 time=9.89 ms
```

```
64 bytes from 183.2.172.42: icmp_seq=3 ttl=53 time=9.64 ms
^C
--- www.a.shifen.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 9.640/9.915/10.219/0.237 ms
```

3. 6. 2. 2. Use nmcli command to set static IP address

1) If you want to set a static IP address for the network port, please plug the network cable into the development board first, and then start setting the static IP address.

2) Then use the **nmcli con show** con show command to view the name of the network device

```
orangepi@orangepir2s:~$ nmcli con show
NAME                UUID                                  TYPE       DEVICE
Wired connection 1  941e80ac-17e7-36e4-9f0e-4ecf7da96763  ethernet   --
Wired connection 2  8ade3ff0-195b-335a-9937-2da136afbadf  ethernet   --
Wired connection 3  df139a10-a820-32bb-8df0-76811709a268  ethernet   --
Wired connection 4  ed9ce37f-bada-3d6f-9379-a4583cbab247  ethernet   --
```

3) Then enter the following command, where

a. **"Wired connection 1"** Indicates setting the static IP address of the Ethernet port. If you need to set the static IP address of other network ports, please modify it to the name corresponding to the corresponding network interface (which can be obtained through the **nmcli con show**)

b. **ipv4.addresses** The following is the static IP address to be set, which can be modified to the value you want to set

c. **ipv4.gateway** Indicates the gateway address

```
orangepi@orangepi:~$ sudo nmcli con mod "Wired connection 1"   \
ipv4.addresses "192.168.1.110/24"   \
ipv4.gateway "192.168.1.1"   \
ipv4.dns "8.8.8.8"   \
ipv4.method "manual"
```

4) Then restart the Linux system

```
orangepi@orangepi:~$ sudo reboot
```

5) Then re-enter the Linux system and use **ip addr show**command to see that the IP address has been set to the desired value.

```
orangepi@orangepi:~$ ip addr show
2: enP1p1s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state
UP group default qlen 1000
    link/ether 00:e0:4c:68:00:26 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.110/24 scope global noprefixroute enP3p49s0
       valid_lft forever preferred_lft forever
    inet6 fe80::9005:95ac:b9c0:2beb/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

3. 7. SSH remote login development board

By default, Linux systems have SSH remote login enabled, and allow the root user to log in to the system. Before SSH login, you must first ensure that the Ethernet or WiFi network is connected, and then use the ip addr command or check the router to obtain the IP address of the development board.

3. 7. 1. SSH remote login to the development board under Ubuntu

1) Get the IP address of the development board.

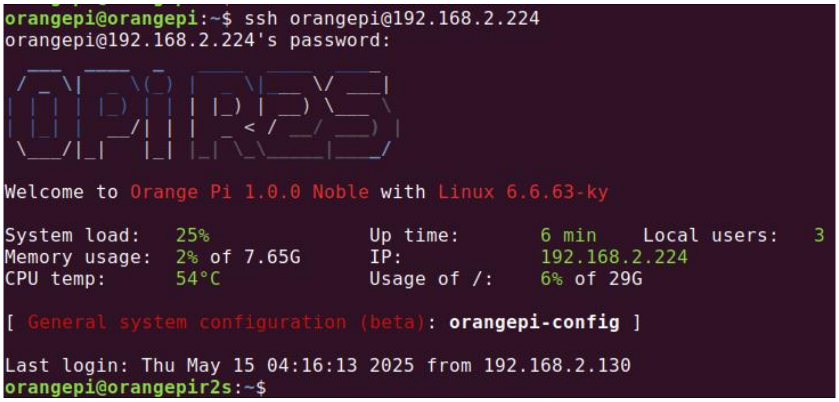
2) Then you can log in to the Linux system remotely through the ssh command.

```
test@test:~$ ssh root@192.168.x.xxx          #Need to be replaced with the IP address
of the development board
root@192.168.x.xx's password:                 #Enter the password here. The default
password is orangepi
```

Note that when you enter the password, the screen will not display the specific content of the password you entered. Please do not think that there is any malfunction. Just press Enter after entering it.

If the prompt refuses to connect, as long as you are using the image provided by Orange Pi, please do not doubt whether the password orangepi is wrong, but look for other reasons.

3) After successfully logging into the system, the display is as shown below:



```
orangepi@orangepi:~$ ssh orangepi@192.168.2.224
orangepi@192.168.2.224's password:
  ___  ____  _   ____  ____  ____
 / _ \|  _ \(_) |  _ \|___ \/ ___|
| | | | |_) | | | |_) | __) \___ \
| |_| |  __/| | |  _ < / __/ ___) |
 \___/|_|   |_| |_| \_\_____|____/

Welcome to Orange Pi 1.0.0 Noble with Linux 6.6.63-ky

System load:   25%              Up time:       6 min     Local users:   3
Memory usage:  2% of 7.65G      IP:            192.168.2.224
CPU temp:      54°C             Usage of /:    6% of 29G

[ General system configuration (beta): orangepi-config ]

Last login: Thu May 15 04:16:13 2025 from 192.168.2.130
orangepi@orangepir2s:~$
```

If ssh cannot log in to the Linux system normally, first check whether the IP address of the development board can be pinged. If the ping is successful, you can log in to the Linux system through the serial port and then enter the following command on the development board to try to connect:

root@orangepi:~# **reset_ssh.sh**

If it still doesn't work, please try burning the system again.

3. 7. 2. SSH remote login development board under Windows

1) First, obtain the IP address of the development board.

2) Under Windows, you can use MobaXterm to remotely log in to the development board. First, create a new ssh session.

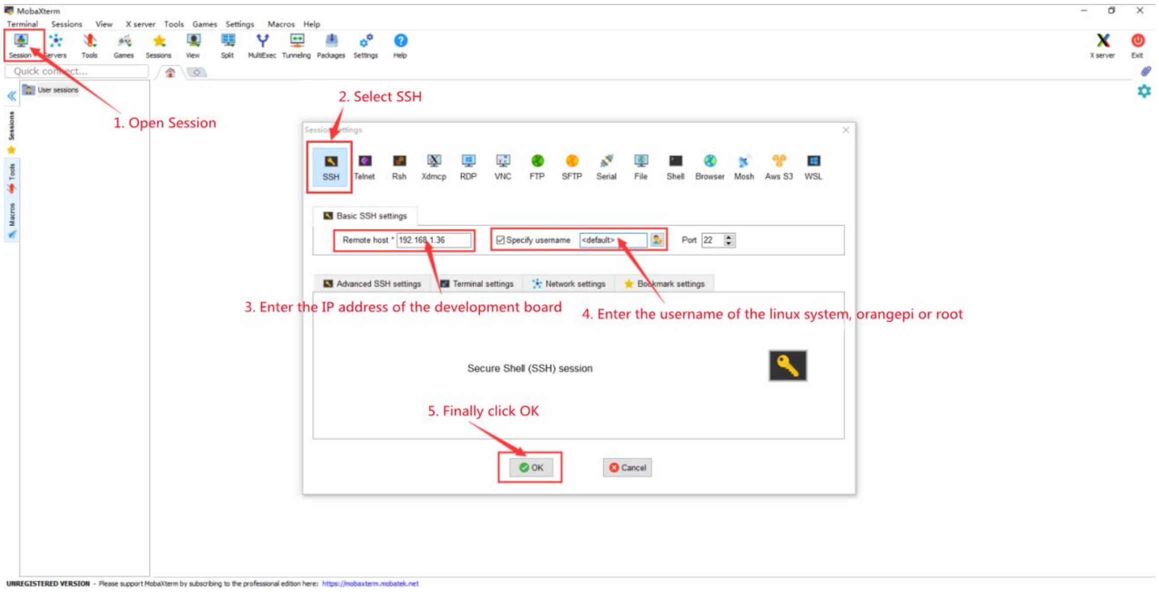
a. Open **Session**.

b. Select **SSH** in **Session Setting**.

c. Enter the IP address of the development board in Remote host.

d. Enter the Linux user name **root** or **root** in **Specify username**.

e. Click **OK**.



1. Open Session

2. Select SSH

Basic SSH settings

Remote host * 192.168.1.36 ☑ Specify username <default> Port 22

3. Enter the IP address of the development board

4. Enter the username of the linux system, orangepi or root

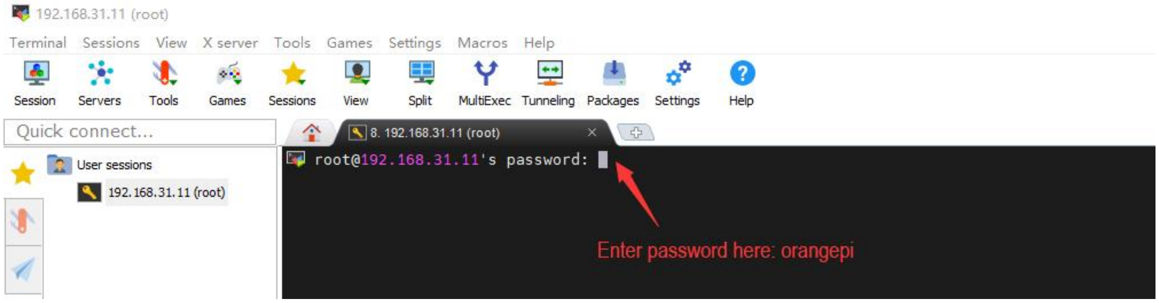
Secure Shell (SSH) session

5. Finally click OK

MobaXterm
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
X server Exit
Quick connect...
User sessions
Session settings
SSH Telnet Rsh Xdmcp RDP VNC FTP SFTP Serial File Shell Browser Mosh Aws S3 WSL
Advanced SSH settings Terminal settings Network settings Bookmark settings
OK Cancel
UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

3) You will then be prompted to enter a password. The default password for both root and orangepi users is orangepi.

Please note that when you enter the password, the specific content of the password will not be displayed on the screen. Please do not think that there is any malfunction. Just press Enter after entering it.

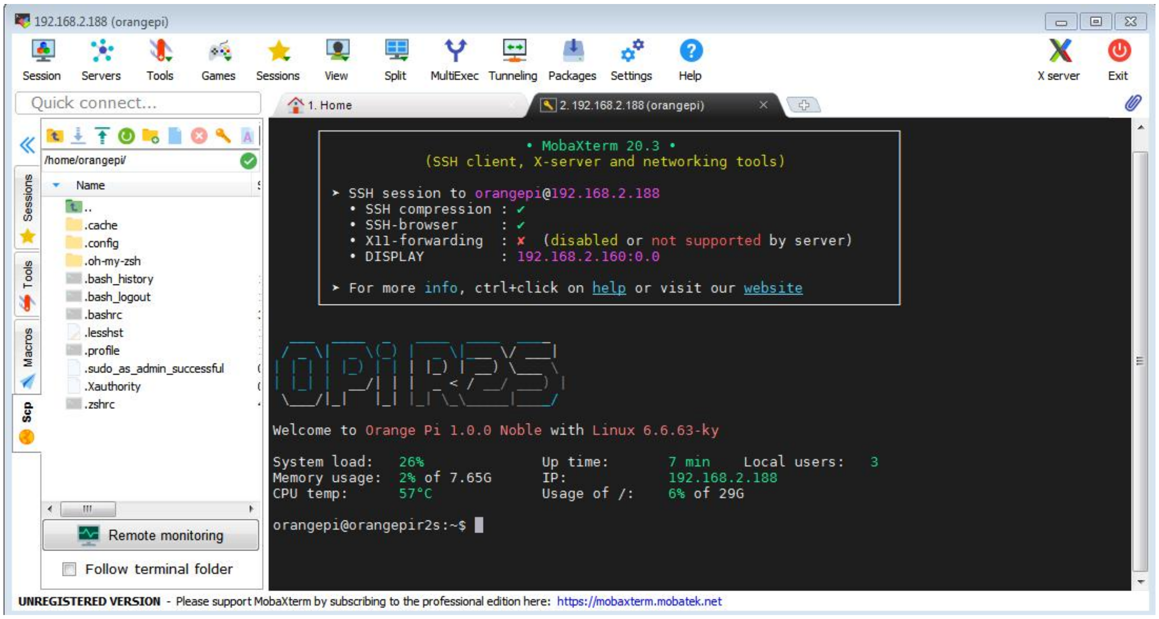


```
root@192.168.31.11's password:
```

Enter password here: orangepi

192.168.31.11 (root)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
Quick connect...
8. 192.168.31.11 (root)
User sessions
192.168.31.11 (root)

4) After successfully logging into the system, the display is as shown below:



```
• MobaXterm 20.3 •
(SSH client, X-server and networking tools)

➤ SSH session to orangepi@192.168.2.188
  • SSH compression : ✔
  • SSH-browser     : ✔
  • X11-forwarding  : ✘  (disabled or not supported by server)
  • DISPLAY         : 192.168.2.160:0.0

➤ For more info, ctrl+click on help or visit our website

Welcome to Orange Pi 1.0.0 Noble with Linux 6.6.63-ky

System load:   26%              Up time:       7 min     Local users:   3
Memory usage:  2% of 7.65G      IP:            192.168.2.188
CPU temp:      57°C             Usage of /:    6% of 29G

orangepi@orangepir2s:~$
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

3. 8. How to upload files to the Linux system of the development board

3. 8. 1. How to upload files from Ubuntu PC to the Linux system of the development board

3. 8. 1. 1. How to upload files using the scp command

1) Use the scp command to upload files from the Ubuntu PC to the Linux system of the development board. The specific commands are as follows:

a. **file_path：** Need to be replaced with the path to the file to be uploaded.

b. **orangepi：** is the user name of the development board's Linux system. It can also be replaced with other names, such as root.

c. **192.168.xx.xx:** It is the IP address of the development board. Please modify it according to the actual situation.

d. **/home/orangepi:**The path in the Linux system of the development board can also be changed to other paths.

```
test@test:~$ scp    file_path    orangepi@192.168.xx.xx:/home/orangepi/
```

2) If you want to upload a folder, you need to add the -r parameter.

```
test@test:~$ scp    -r    dir_path    orangepi@192.168.xx.xx:/home/orangepi/
```

3) There are more uses for scp. Please use the following command to view the man manual.

```
test@test:~$ man scp
```

3. 8. 1. 2. How to upload files using filezilla

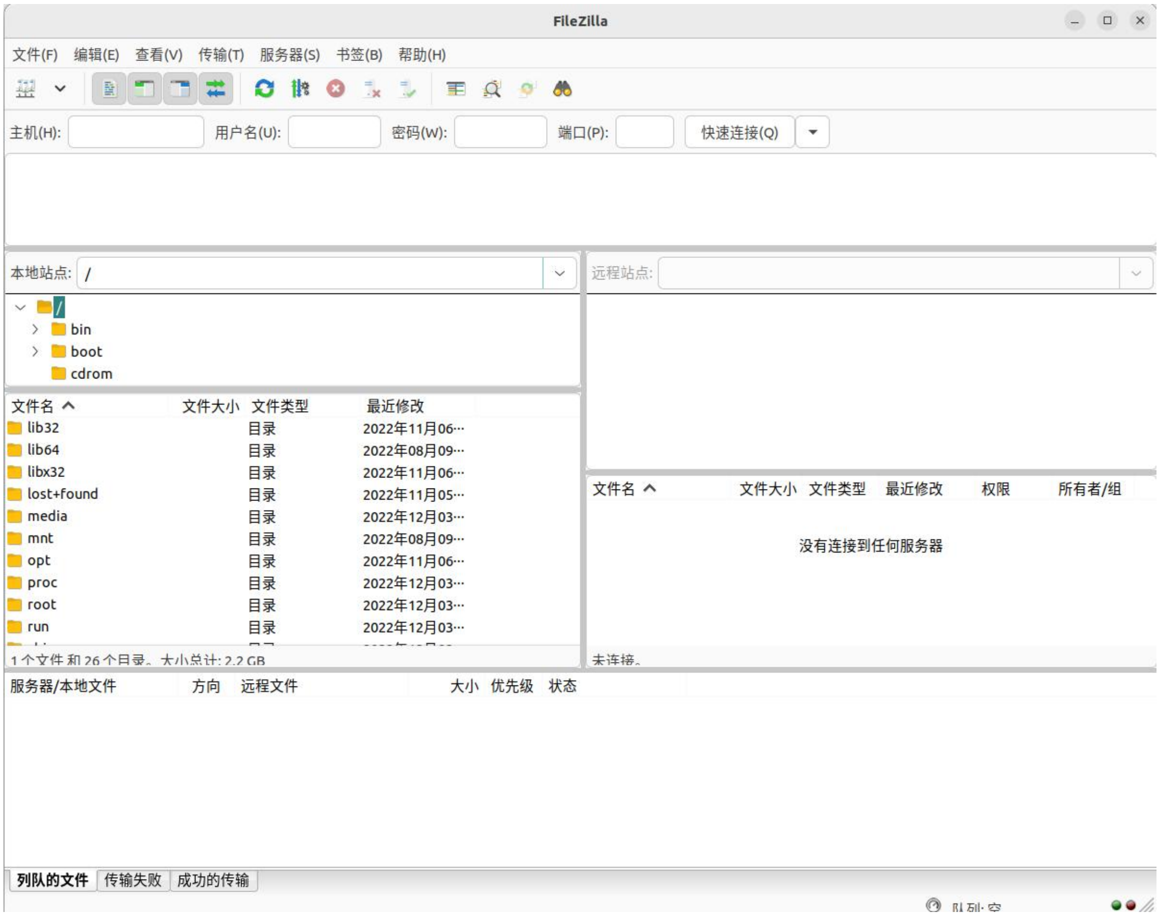
1) First install filezilla in your Ubuntu PC.

```
test@test:~$ sudo apt install -y filezilla
```

2) Then open filezilla using the command below.

```
test@test:~$ filezilla
```

3) The interface after opening filezilla is as shown below. At this time, the remote site on the right is empty.


FileZilla
文件(F) 编辑(E) 查看(V) 传输(T) 服务器(S) 书签(B) 帮助(H)
主机(H): 用户名(U): 密码(W): 端口(P): 快速连接(Q)
本地站点: /
远程站点:
/
bin
boot
cdrom
列队的文件 传输失败 成功的传输
队列: 空

文件名	文件大小	文件类型	最近修改
lib32		目录	2022年11月06···
lib64		目录	2022年08月09···
libx32		目录	2022年11月06···
lost+found		目录	2022年11月05···
media		目录	2022年12月03···
mnt		目录	2022年08月09···
opt		目录	2022年11月06···
proc		目录	2022年12月03···
root		目录	2022年12月03···
run		目录	2022年12月03···

1 个文件 和 26 个目录。大小总计: 2.2 GB

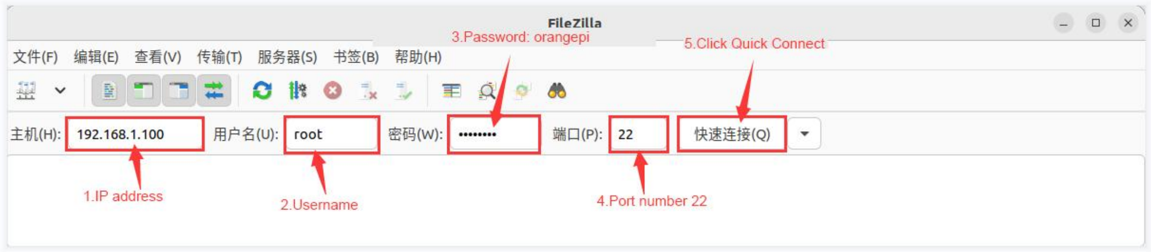
文件名	文件大小	文件类型	最近修改	权限	所有者/组

没有连接到任何服务器

未连接。

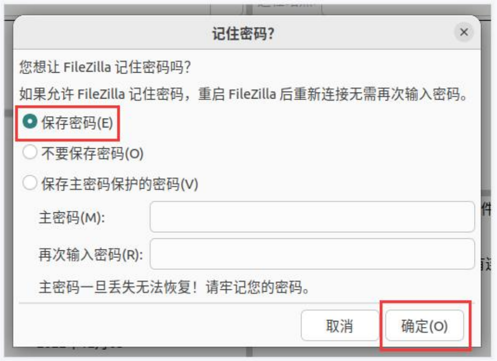
服务器/本地文件	方向	远程文件	大小	优先级	状态

4) The method of connecting the development board is shown in the figure below:



主机(H): 192.168.1.100 用户名(U): root 密码(W): •••••••• 端口(P): 22 快速连接(Q)

5) Then select **Save Password** and click **OK**.



记住密码?

您想让 FileZilla 记住密码吗?

如果允许 FileZilla 记住密码，重启 FileZilla 后重新连接无需再次输入密码。

● 保存密码(E)

○ 不要保存密码(O)

○ 保存主密码保护的密码(V)

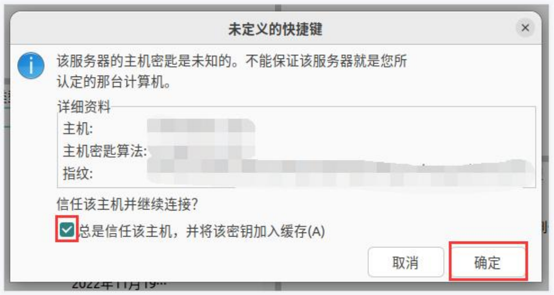
主密码(M):

再次输入密码(R):

主密码一旦丢失无法恢复！请牢记您的密码。

取消 确定(O)

6) Then select **Always trust this host** and click **OK.**



未定义的快捷键

该服务器的主机密匙是未知的。不能保证该服务器就是您所认定的那台计算机。

详细资料

主机: [redacted]

主机密匙算法:

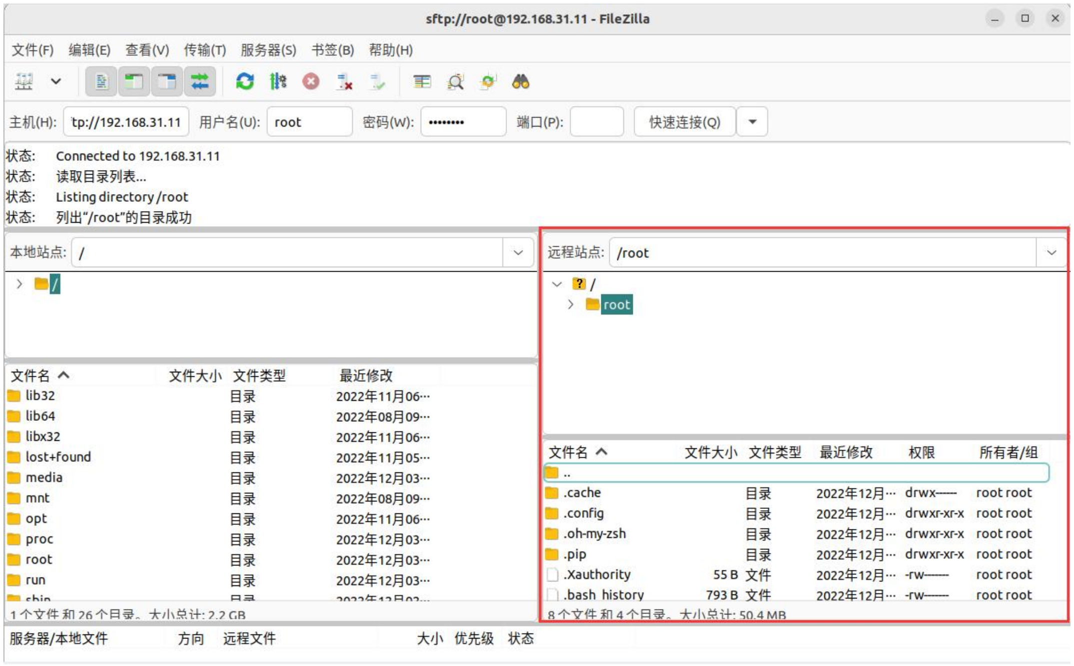
指纹:

信任该主机并继续连接?

☑ 总是信任该主机，并将该密钥加入缓存(A)

取消 确定

7) After the connection is successful, you can see the directory structure of the development board's Linux file system on the right side of the filezilla software.


sftp://root@192.168.31.11 - FileZilla
文件(F) 编辑(E) 查看(V) 传输(T) 服务器(S) 书签(B) 帮助(H)
快速连接(Q)
服务器/本地文件 方向 远程文件 大小 优先级 状态

主机(H): tp://192.168.31.11 用户名(U): root 密码(W): •••••••• 端口(P):

```
状态:   Connected to 192.168.31.11
状态:   读取目录列表...
状态:   Listing directory /root
状态:   列出"/root"的目录成功
```

本地站点: /

文件名	文件大小	文件类型	最近修改
lib32		目录	2022年11月06…
lib64		目录	2022年08月09…
libx32		目录	2022年11月06…
lost+found		目录	2022年11月05…
media		目录	2022年12月03…
mnt		目录	2022年08月09…
opt		目录	2022年11月06…
proc		目录	2022年12月03…
root		目录	2022年12月03…
run		目录	2022年12月03…
sbin		目录	2022年12月03…

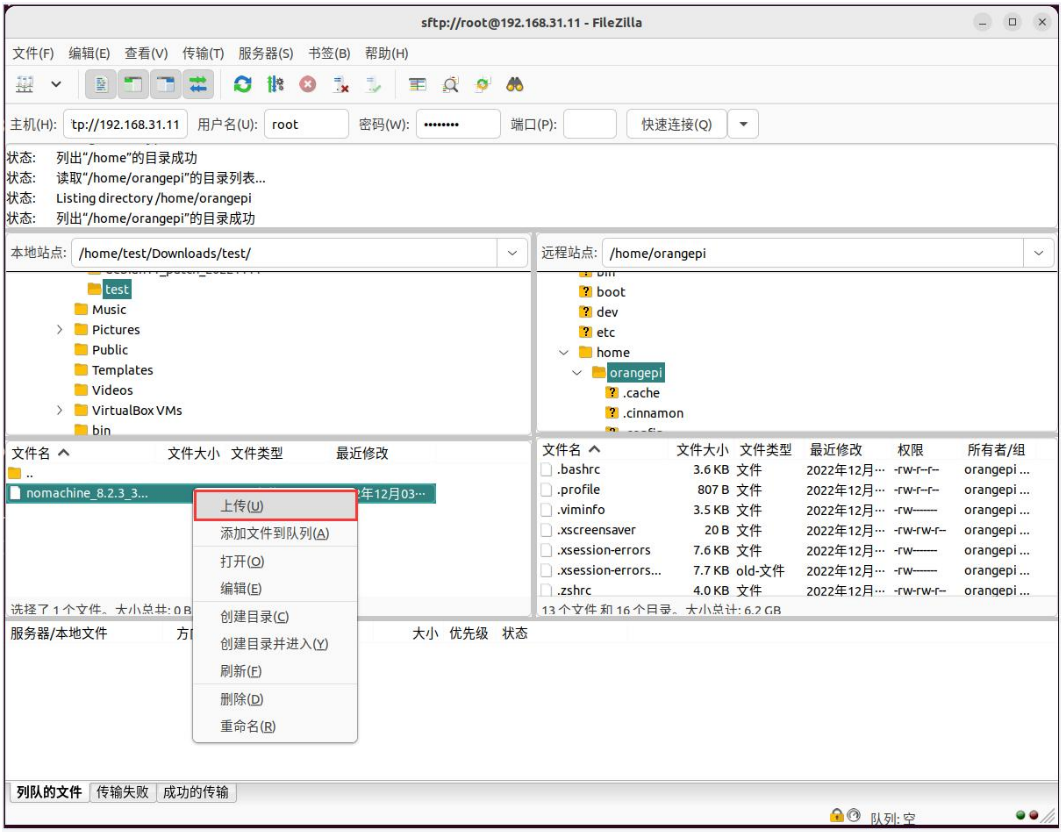
1 个文件 和 26 个目录。大小总计: 2.2 GB

远程站点: /root

文件名	文件大小	文件类型	最近修改	权限	所有者/组
..					
.cache		目录	2022年12月…	drwx------	root root
.config		目录	2022年12月…	drwxr-xr-x	root root
.oh-my-zsh		目录	2022年12月…	drwxr-xr-x	root root
.pip		目录	2022年12月…	drwxr-xr-x	root root
.Xauthority	55 B	文件	2022年12月…	-rw-------	root root
.bash_history	793 B	文件	2022年12月…	-rw-------	root root

8 个文件 和 4 个目录。大小总计: 50.4 MB

8) Then select the path to be uploaded to the development board on the right side of the filezilla software, then select the file to be uploaded in the Ubuntu PC on the left side of the filezilla software, right-click the mouse, and then click the upload option to start uploading the file to the development board.



sftp://root@192.168.31.11 - FileZilla

主机(H): tp://192.168.31.11 用户名(U): root 密码(W): •••••••• 端口(P):

```
状态:   列出"/home"的目录成功
状态:   读取"/home/orangepi"的目录列表...
状态:   Listing directory /home/orangepi
状态:   列出"/home/orangepi"的目录成功
```

本地站点: /home/test/Downloads/test/

远程站点: /home/orangepi

文件名	文件大小	文件类型	最近修改
..			
nomachine_8.2.3_3...			年12月03···

选择了 1 个文件。大小总计: 0 B

文件名	文件大小	文件类型	最近修改	权限	所有者/组
.bashrc	3.6 KB	文件	2022年12月···	-rw-r--r--	orangepi ...
.profile	807 B	文件	2022年12月···	-rw-r--r--	orangepi ...
.viminfo	3.5 KB	文件	2022年12月···	-rw-------	orangepi ...
.xscreensaver	20 B	文件	2022年12月···	-rw-rw-r--	orangepi ...
.xsession-errors	7.6 KB	文件	2022年12月···	-rw-------	orangepi ...
.xsession-errors...	7.7 KB	old-文件	2022年12月···	-rw-------	orangepi ...
.zshrc	4.0 KB	文件	2022年12月···	-rw-rw-r--	orangepi ...

13 个文件 和 16 个目录。大小总计: 6.2 GB

队列: 空

9) After uploading is complete, you can go to the corresponding path in the Linux system of the development board to view the uploaded files.

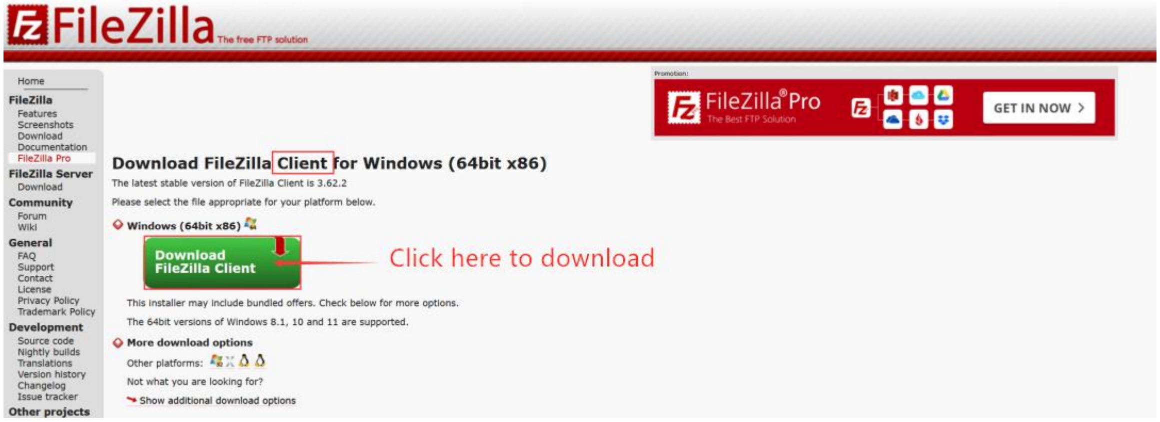
10) The method for uploading a folder is the same as that for uploading a file, so I will not go into details here.

3. 8. 2. How to upload files from Windows PC to the Linux system of the development board

3. 8. 2. 1. How to upload files using FileZilla

1) First download the installation file of the Windows version of the filezilla software. The download link is as follows:

https://filezilla-project.org/download.php?type=client



Download FileZilla Client for Windows (64bit x86)

The latest stable version of FileZilla Client is 3.62.2

Please select the file appropriate for your platform below.

Windows (64bit x86)

This installer may include bundled offers. Check below for more options.

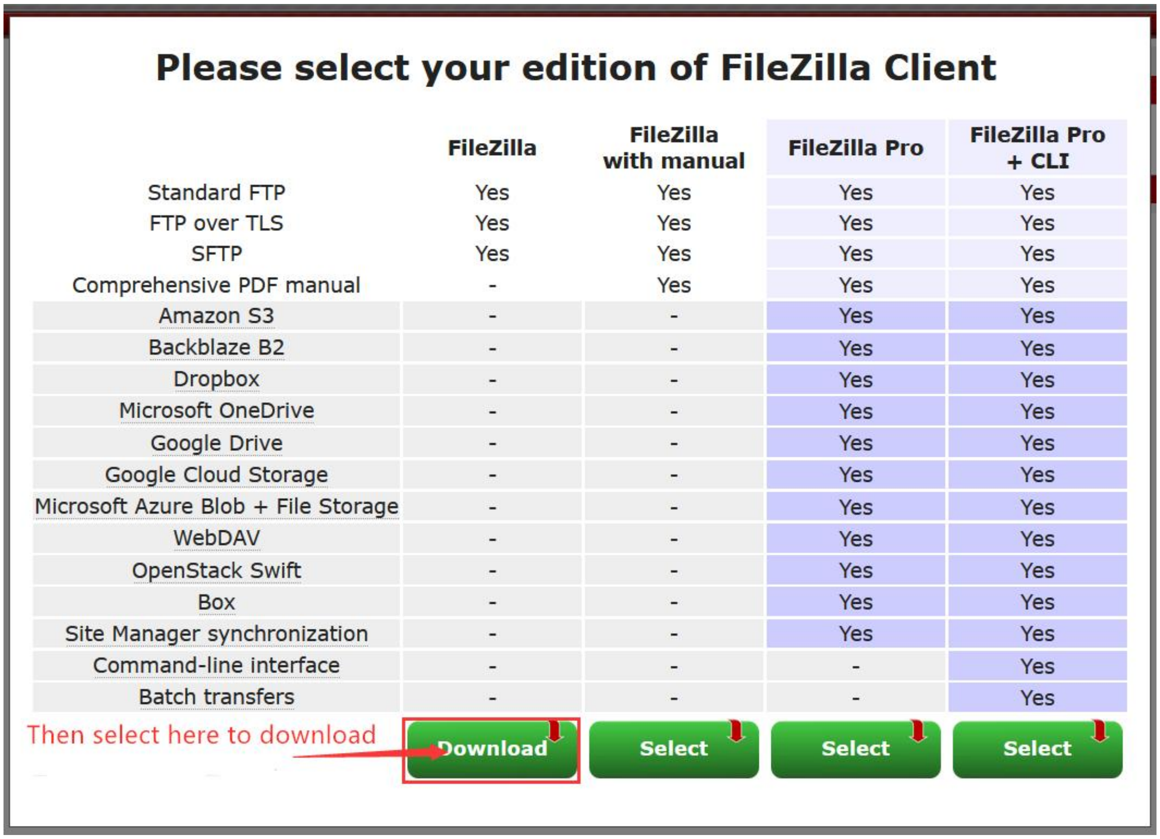
The 64bit versions of Windows 8.1, 10 and 11 are supported.

More download options

Other platforms:

Not what you are looking for?

Show additional download options



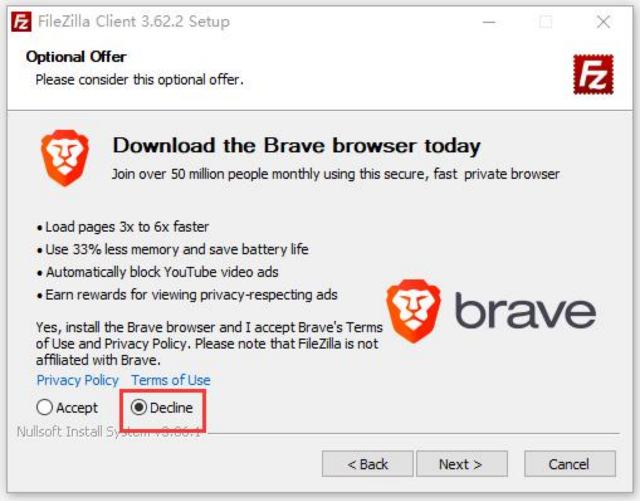
Please select your edition of FileZilla Client

	FileZilla	FileZilla with manual	FileZilla Pro	FileZilla Pro + CLI
Standard FTP	Yes	Yes	Yes	Yes
FTP over TLS	Yes	Yes	Yes	Yes
SFTP	Yes	Yes	Yes	Yes
Comprehensive PDF manual	-	Yes	Yes	Yes
Amazon S3	-	-	Yes	Yes
Backblaze B2	-	-	Yes	Yes
Dropbox	-	-	Yes	Yes
Microsoft OneDrive	-	-	Yes	Yes
Google Drive	-	-	Yes	Yes
Google Cloud Storage	-	-	Yes	Yes
Microsoft Azure Blob + File Storage	-	-	Yes	Yes
WebDAV	-	-	Yes	Yes
OpenStack Swift	-	-	Yes	Yes
Box	-	-	Yes	Yes
Site Manager synchronization	-	-	Yes	Yes
Command-line interface	-	-	-	Yes
Batch transfers	-	-	-	Yes

2) The downloaded installation package is as shown below, and you can double-click it to install it directly.

```
FileZilla_Server_1.5.1_win64-setup.exe
```

During the installation process, select `Decline` on the following installation interface, and then select `Next>.`



Optional Offer

Please consider this optional offer.

Download the Brave browser today

Join over 50 million people monthly using this secure, fast private browser

- Load pages 3x to 6x faster
- Use 33% less memory and save battery life
- Automatically block YouTube video ads
- Earn rewards for viewing privacy-respecting ads

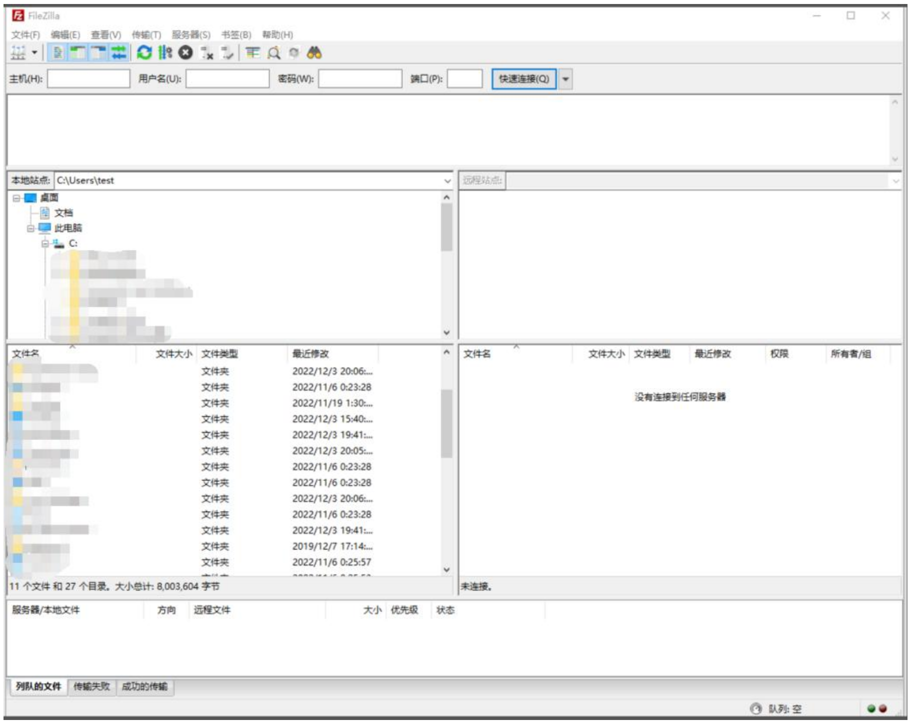
Yes, install the Brave browser and I accept Brave's Terms of Use and Privacy Policy. Please note that FileZilla is not affiliated with Brave.

Privacy Policy Terms of Use

○ Accept ● Decline

Nullsoft Install System v3.08.1

3) The interface after opening filezilla is as shown below. At this time, the remote site on the right is empty.



主机(H): 用户名(U): 密码(W): 端口(P):

本地站点: C:\Users\test

远程站点:

文件名	文件大小	文件类型	最近修改
		文件夹	2022/12/3 20:06:...
		文件夹	2022/11/6 0:23:28
		文件夹	2022/11/19 1:30:...
		文件夹	2022/12/3 15:40:...
		文件夹	2022/12/3 19:41:...
		文件夹	2022/12/3 20:05:...
		文件夹	2022/11/6 0:23:28
		文件夹	2022/11/6 0:23:28
		文件夹	2022/12/3 20:06:...
		文件夹	2022/11/6 0:23:28
		文件夹	2022/12/3 19:41:...
		文件夹	2019/12/7 17:14:...
		文件夹	2022/11/6 0:25:57

11 个文件 和 27 个目录。大小总计: 8,003,604 字节

文件名	文件大小	文件类型	最近修改	权限	所有者/组

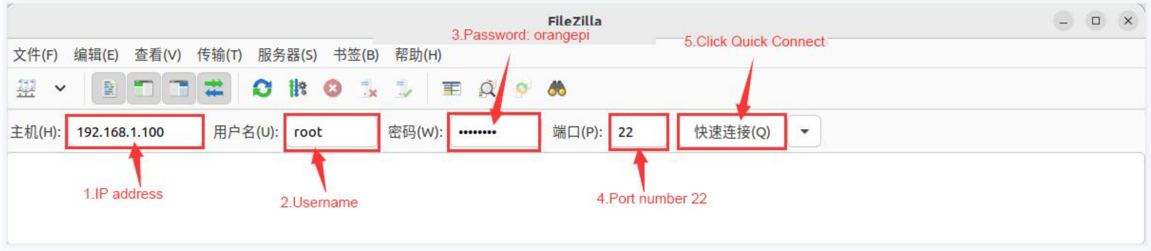
没有连接到任何服务器

未连接。

服务器/本地文件	方向	远程文件	大小	优先级	状态

队列: 空

4) The method of connecting the development board is shown in the figure below:



主机(H): 192.168.1.100 用户名(U): root 密码(W): •••••••• 端口(P): 22 快速连接(Q)

1.IP address

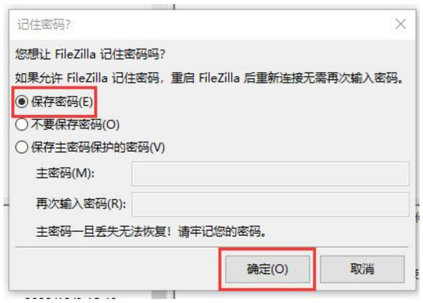
2.Username

3.Password: orangepi

4.Port number 22

5.Click Quick Connect

5) Then select **Save Password** and click **OK.**



记住密码?

您想让 FileZilla 记住密码吗?

如果允许 FileZilla 记住密码，重启 FileZilla 后重新连接无需再次输入密码。

● 保存密码(E)
○ 不要保存密码(O)
○ 保存主密码保护的密码(V)

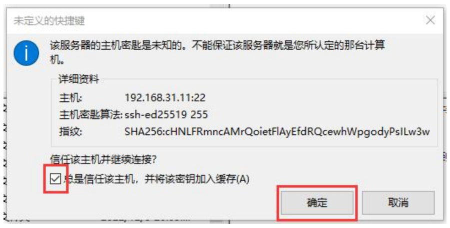
主密码(M):

再次输入密码(R):

主密码一旦丢失无法恢复！请牢记您的密码。

确定(O) 取消

6) Then select **Always trust this host** and click **OK.**



未定义的快捷键

该服务器的主机密匙是未知的。不能保证该服务器就是您所认定的那台计算机。

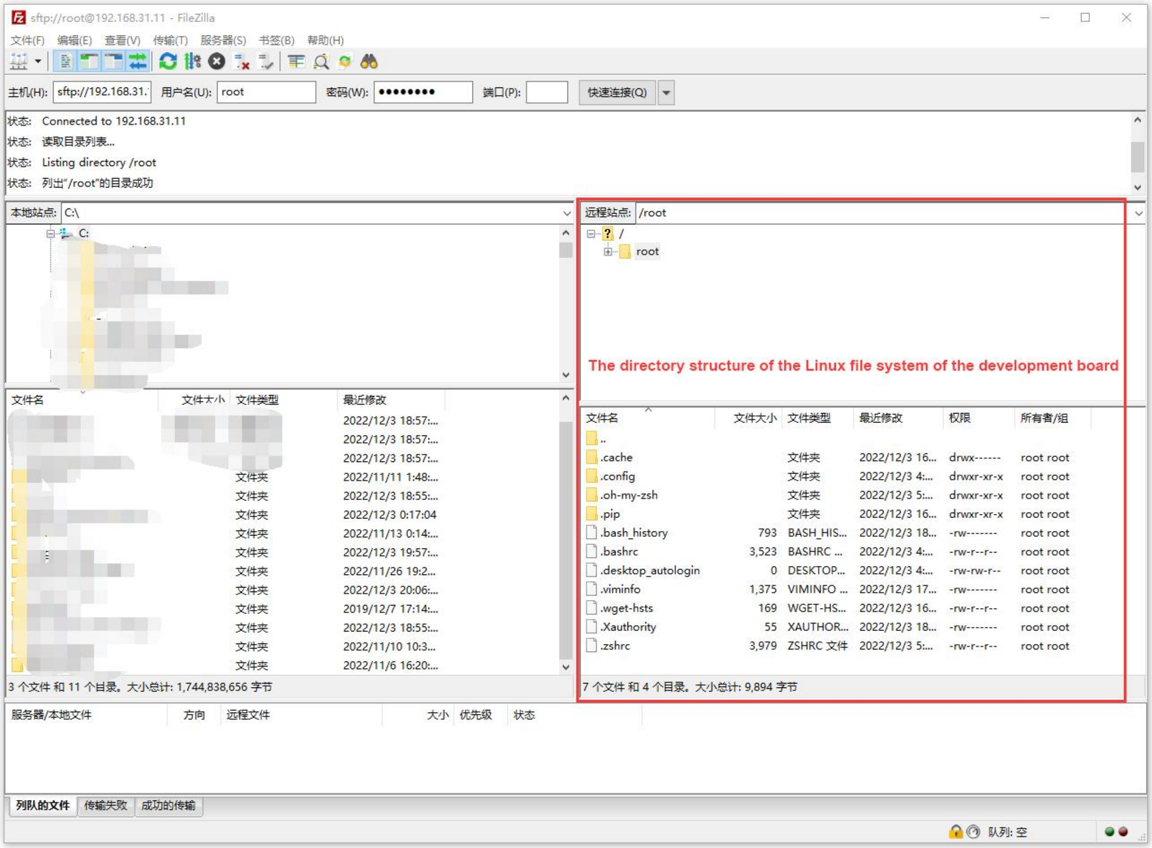
详细资料
主机: 192.168.31.11:22
主机密匙算法: ssh-ed25519 255
指纹: SHA256:cHNLFRmncAMrQoietFlAyEfdRQcewhWpgodyPsILw3w

信任该主机并继续连接?

☑ 总是信任该主机，并将该密钥加入缓存(A)

确定 取消

7) After the connection is successful, you can see the directory structure of the development board's Linux file system on the right side of the filezilla software.


sftp://root@192.168.31.11 - FileZilla
文件(F) 编辑(E) 查看(V) 传输(T) 服务器(S) 书签(B) 帮助(H)
快速连接(Q)
服务器/本地文件 方向 远程文件 大小 优先级 状态
列队的文件 传输失败 成功的传输
队列: 空

主机(H): sftp://192.168.31. 用户名(U): root 密码(W): •••••••• 端口(P):

```
状态:   Connected to 192.168.31.11
状态:   读取目录列表...
状态:   Listing directory /root
状态:   列出"/root"的目录成功
```

本地站点: C:\

C:

文件名	文件大小	文件类型	最近修改
			2022/12/3 18:57:...
			2022/12/3 18:57:...
			2022/12/3 18:57:...
		文件夹	2022/11/11 1:48:...
		文件夹	2022/12/3 18:55:...
		文件夹	2022/12/3 0:17:04
		文件夹	2022/11/13 0:14:...
		文件夹	2022/12/3 19:57:...
		文件夹	2022/11/26 19:2...
		文件夹	2022/12/3 20:06:...
		文件夹	2019/12/7 17:14:...
		文件夹	2022/12/3 18:55:...
		文件夹	2022/11/10 10:3...
		文件夹	2022/11/6 16:20:...

3 个文件 和 11 个目录。大小总计: 1,744,838,656 字节

远程站点: /root

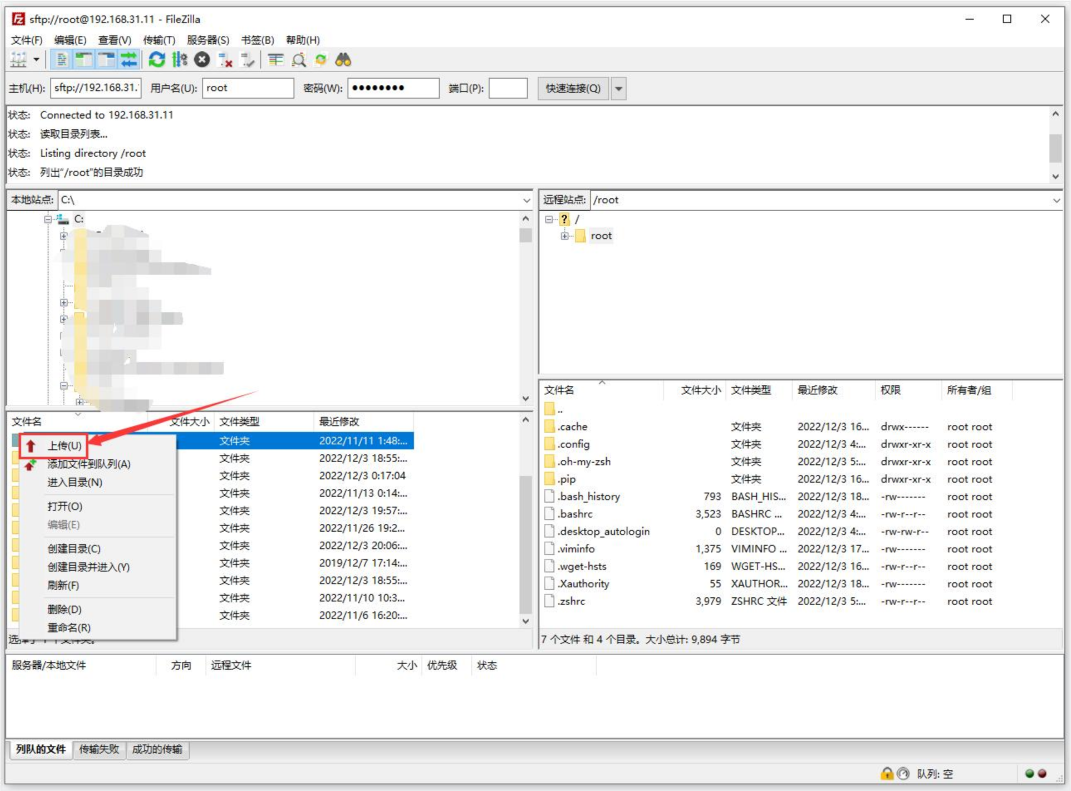
/
root

The directory structure of the Linux file system of the development board

文件名	文件大小	文件类型	最近修改	权限	所有者/组
..					
.cache		文件夹	2022/12/3 16...	drwx------	root root
.config		文件夹	2022/12/3 4:...	drwxr-xr-x	root root
.oh-my-zsh		文件夹	2022/12/3 5:...	drwxr-xr-x	root root
.pip		文件夹	2022/12/3 16...	drwxr-xr-x	root root
.bash_history	793	BASH_HIS...	2022/12/3 18...	-rw-------	root root
.bashrc	3,523	BASHRC ...	2022/12/3 4:...	-rw-r--r--	root root
.desktop_autologin	0	DESKTOP...	2022/12/3 4:...	-rw-rw-r--	root root
.viminfo	1,375	VIMINFO ...	2022/12/3 17...	-rw-------	root root
.wget-hsts	169	WGET-HS...	2022/12/3 16...	-rw-r--r--	root root
.Xauthority	55	XAUTHOR...	2022/12/3 18...	-rw-------	root root
.zshrc	3,979	ZSHRC 文件	2022/12/3 5:...	-rw-r--r--	root root

7 个文件 和 4 个目录。大小总计: 9,894 字节

8) Then select the path to be uploaded to the development board on the right side of the filezilla software, then select the file to be uploaded in the Windows PC on the left side of the filezilla software, right-click the mouse, and then click the upload option to start uploading the file to the development board.



主机(H): sftp://192.168.31.　用户名(U): root　密码(W): ●●●●●●●●●　端口(P):

```
状态:	Connected to 192.168.31.11
状态:	读取目录列表...
状态:	Listing directory /root
状态:	列出"/root"的目录成功
```

本地站点: C:\

远程站点: /root

文件名	文件大小	文件类型	最近修改
		文件夹	2022/11/11 1:48:...
		文件夹	2022/12/3 18:55:...
		文件夹	2022/12/3 0:17:04
		文件夹	2022/11/13 0:14:...
		文件夹	2022/12/3 19:57:...
		文件夹	2022/11/26 19:2...
		文件夹	2022/12/3 20:06:...
		文件夹	2019/12/7 17:14:...
		文件夹	2022/12/3 18:55:...
		文件夹	2022/11/10 10:3...
		文件夹	2022/11/6 16:20:...

文件名	文件大小	文件类型	最近修改	权限	所有者/组
..					
.cache		文件夹	2022/12/3 16...	drwx------	root root
.config		文件夹	2022/12/3 4:...	drwxr-xr-x	root root
.oh-my-zsh		文件夹	2022/12/3 5:...	drwxr-xr-x	root root
.pip		文件夹	2022/12/3 16...	drwxr-xr-x	root root
.bash_history	793	BASH_HIS...	2022/12/3 18...	-rw-------	root root
.bashrc	3,523	BASHRC ...	2022/12/3 4:...	-rw-r--r--	root root
.desktop_autologin	0	DESKTOP...	2022/12/3 4:...	-rw-rw-r--	root root
.viminfo	1,375	VIMINFO ...	2022/12/3 17...	-rw-------	root root
.wget-hsts	169	WGET-HS...	2022/12/3 16...	-rw-r--r--	root root
.Xauthority	55	XAUTHOR...	2022/12/3 18...	-rw-------	root root
.zshrc	3,979	ZSHRC 文件	2022/12/3 5:...	-rw-r--r--	root root

7 个文件 和 4 个目录。大小总计: 9,894 字节

9) After uploading is complete, you can go to the corresponding path in the Linux system of the development board to view the uploaded files.

10) The method for uploading a folder is the same as that for uploading a file, so I will not go into details here.

3. 9. USB interface test

The USB port can be connected to a USB hub to expand the number of USB ports.

3. 9. 1. Test by connecting USB storage device

1) First, insert the USB flash drive or USB mobile hard disk into the USB port of the Orange Pi development board.

2) Execute the following command. If you can see the output of sdX, it means that the USB disk has been successfully recognized.

```
orangepi@orangepi:~$ cat /proc/partitions | grep "sd*"
```

```
major minor   #blocks   name
    8           0      30044160 sda
    8           1      30043119 sda1
```

3) Use the mount command to mount the USB drive to **/mnt**, and then you can view the files in the USB drive.

```
orangepi@orangepi:~$ sudo mount /dev/sda1 /mnt/
orangepi@orangepi:~$ ls /mnt/
test.txt
```

4) After mounting, you can use the **df -h** command to view the capacity usage and mount point of the USB drive.

```
orangepi@orangepi:~$ df -h | grep "sd"
/dev/sda1               29G    208K      29G       1% /mnt
```

3. 9. 2. USB camera test

1) First, you need to prepare a USB camera that supports UVC protocol as shown in the figure below or similar, and then insert the USB camera into the USB port of the Orange Pi development board.

2) Through the v4l2-ctl command, you can see that the device node information of the USB camera is **/dev/video20**

```
orangepi@orangepi:~$ v4l2-ctl --list-devices | grep -A 3 "Q8 HD Webcam"
Q8 HD Webcam: Q8 HD Webcam (usb-fc880000.usb-1):
            /dev/video20
            /dev/video21
            /dev/media1
```

Note that the l in v4l2 is a lowercase letter l, not the number 1.

In addition, the serial number of the video is not necessarily video20, please refer to the actual one you see.

3) How to use fswebcam to test USB camera

a. Install fswebcam

```
orangepi@orangepi:~$ sudo apt update
orangepi@orangepi:~$ sudo apt-get install -y fswebcam
```

b. After installing fswebcam, you can use the following command to take photos

a) The -d option is used to specify the device node of the USB camera

b) --no-banner is used to remove the watermark of the photo

c) The -r option is used to specify the resolution of the photo.

d) -S option is used to set the number of frames to skip ahead

e) ./image.jpg Used to set the name and path of the generated photo

```
orangepi@orangepi:~$ sudo fswebcam -d /dev/video20   \
--no-banner -r 1280x720 -S 5 ./image.jpg
```

c. In the server version of Linux, after taking the photo, you can use the scp command to transfer the photo to the Ubuntu PC for mirror viewing.

```
orangepi@orangepi:~$ scp image.jpg test@192.168.1.55:/home/test （Modify the IP
address and path according to the actual situation）
```

3. 10. Temperature sensor

1) The command to view the system temperature sensor is:

```
orangepi@orangepi:~$ sensors
cluster0_thermal-virtual-0
Adapter: Virtual device
temp1:         +59.0°C

cluster1_thermal-virtual-0
Adapter: Virtual device
temp1:         +60.0°C
```

3. 11. Hardware watchdog test

The Linux system released by Orange Pi has the watchdog_test program pre-installed, which can be used for direct testing.

The method to run the watchdog_test program is as follows:

a. The second parameter 10 represents the watchdog counting time. If the watchdog is not fed within this time, the system will restart.

b. We can feed the dog by pressing any key on the keyboard (except ESC). After feeding the dog, the program will print a line of keep alive to indicate that the dog was successfully fed.

```
orangepi@orangepi:~$ sudo watchdog_test 10
open success
options is 33152,identity is sunxi-wdt
put_usr return,if 0,success:0
The old reset time is: 16
return ENOTTY,if -1,success:0
return ENOTTY,if -1,success:0
put_user return,if 0,success:0
put_usr return,if 0,success:0
keep alive
keep alive
keep alive
```

3. 12. Test of some programming languages supported by Linux system

3. 12. 1. Ubuntu Noble System

1) Ubuntu Jammy is installed with the gcc compilation tool chain by default, which can compile C language programs directly in the Linux system of the development board

a. gcc version is as follows

```
orangepi@orangepi:~$ gcc --version
gcc (Ubuntu 13.3.0-6ubuntu2~24.04) 13.3.0
Copyright (C) 2023 Free Software Foundation, Inc.
```

```
This is free software; see the source for copying conditions.   There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR
PURPOSE.
```

b. Write the **hello_world.c** program in C language

```
orangepi@orangepi:~$ vim hello_world.c
#include <stdio.h>

int main(void)
{
        printf("Hello World!\n");

        return 0;
}
```

c. Then compile and run **hello_world.c**

```
orangepi@orangepi:~$ gcc -o hello_world hello_world.c
orangepi@orangepi:~$ ./hello_world
Hello World!
```

2) Ubuntu Jammy has Python 3 installed by default

a. The specific version of Python3 is as follows

```
orangepi@orangepi:~$ python3
Python 3.12.3 (main, Nov   6 2024, 18:32:19) [GCC 13.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.

>>>
```

b. Write the **hello_world.py** program in Python

```
orangepi@orangepi:~$ vim hello_world.py
print('Hello World!')
```

c. The result of running **hello_world.py** is as follows

```
orangepi@orangepi:~$ python3 hello_world.py
Hello World!
```

3) Ubuntu Noble does not install Java compilation tools and runtime environment by default

a. You can use the following command to install openjdk-21

```
orangepi@orangepi:~$ sudo apt install -y openjdk-21-jdk
```

b. After installation, you can check the Java version

```
orangepi@orangepi:~$ java --version
openjdk 21.0.7 2025-04-15
OpenJDK Runtime Environment (build 21.0.7+6-Ubuntu-1ubuntu124.04)
OpenJDK 64-Bit Server VM (build 21.0.7+6-Ubuntu-1ubuntu124.04, mixed mode,
sharing)
```

c. Write the Java version of **hello_world.java**

```
orangepi@orangepi:~$ vim hello_world.java
public class hello_world
{
        public static void main(String[] args)
        {
                System.out.println("Hello World!");
        }
}
```

d. Then compile and run **hello_world.java**

```
orangepi@orangepi:~$ javac hello_world.java
orangepi@orangepi:~$ java hello_world
Hello World!
```

3. 13. How to install kernel header files

1) The Linux image released by OPi comes with a deb package of kernel header files by default, which is stored in **/opt/**

> **The names of the deb packages of kernel header files of different kernel versions may be different. Please refer to the actual ones you see.**

```
orangepi@orangepi:~$ ls /opt/linux-headers*
/opt/linux-headers-current-ky_x.x.x_riscv64.deb
```

2) Use the following command to install the kernel header file deb package

> **The name of the kernel header file deb package needs to be replaced with the**

actual name, please do not copy it.

```
orangepi@orangepi:~$ sudo dpkg -i /opt/linux-headers-current-ky_1.x.x_riscv64.deb
```

3) After installation, you can see the folder where the kernel header files are located under **/usr/src**

```
orangepi@orangepi:~$ ls /usr/src
linux-headers-6.6.36-ky
```

4) Then you can write a hello kernel module to test the kernel header file

a. First, write the code for the hello kernel module as follows:

```
orangepi@orangepi:~$ vim hello.c
#include <linux/init.h>
#include <linux/module.h>

static int hello_init(void)
{
        printk("Hello Orange Pi -- init\n");

        return 0;
}
static void hello_exit(void)
{
        printk("Hello Orange Pi -- exit\n");

        return;
}

module_init(hello_init);
module_exit(hello_exit);

MODULE_LICENSE("GPL");
```

b. Then write the Makefile file to compile the hello kernel module as follows:

```
orangepi@orangepi:~$ vim Makefile
ifneq ($(KERNELRELEASE),)
obj-m:=hello.o
```

```
else
KDIR :=/lib/modules/$(shell uname -r)/build
PWD   :=$(shell pwd)
all:
    make -C $(KDIR) M=$(PWD) modules
clean:
    rm -f *.ko *.o *.mod.o *.mod *.symvers *.cmd   *.mod.c *.order
endif
```

c. Then use the make command to compile the hello kernel module. The output of the compilation process is as follows:

If there is a problem compiling the copied code here, you can directly use the source code pre-installed in the Linux system. The path of the hello source code is: /usr/src/hello.

```
orangepi@orangepi:~$ sudo make
make -C /lib/modules/6.6.36-ky/build M=/home/orangepi modules
make[1]: Entering directory '/usr/src/linux-headers-6.6.36-ky'
  CC [M]   /home/orangepi/hello.o
  MODPOST /home/orangepi/Module.symvers
  CC [M]   /home/orangepi/hello.mod.o
  LD [M]   /home/orangepi/hello.ko
make[1]: Leaving directory '/usr/src/linux-headers-6.6.36-ky'
```

d. After compilation, the **hello.ko** kernel module will be generated

```
orangepi@orangepi:~$ ls *.ko
hello.ko
```

e. Use the **insmod** command to insert the **hello.ko** kernel module into the kernel

```
orangepi@orangepi:~$ sudo insmod hello.ko
```

f. Then use the **demsg** command to view the output of the **hello.ko** kernel module. If you can see the following output, it means that the **hello.ko** kernel module is loaded correctly.

```
orangepi@orangepi:~$ sudo dmesg | grep "Hello"
[ 2871.893988] Hello Orange Pi -- init
```

g. Use the **rmmod** command to uninstall the **hello.ko** kernel module

```
orangepi@orangepi:~$ sudo rmmod hello
orangepi@orangepi:~$ sudo dmesg | grep "Hello"
[ 2871.893988] Hello Orange Pi -- init
[ 3173.800892] Hello Orange Pi -- exit
```

3. 14. How to shut down and restart the development board

1) When the Linux system is running, if you unplug the Type-C power directly to cut off the power, the file system may lose some data or be damaged. Therefore, please use the **poweroff** command to shut down the Linux system of the development board before unplugging the power.

```
orangepi@orangepi:~$ sudo poweroff
```

2) The command to restart the Linux system is:

```
orangepi@orangepi:~$ sudo reboot
```

4. Linux SDK——Instructions for using orangepi-build

4. 1. Compilation system requirements

1) Linux SDK, namely **orangepi-build**, supports running on computers with **Ubuntu 22.04** installed, so before downloading orangepi-build, please first make sure that the Ubuntu version installed on your computer is Ubuntu **22.04**. The command to check the Ubuntu version installed on the computer is as follows. If the Release field does not display 22.04, it means that the current Ubuntu version does not meet the requirements. Please change the system before performing the following operations.

```
test@test:~$ lsb_release -a
No LSB modules are available.
Distributor ID:    Ubuntu
Description:       Ubuntu 22.04 LTS
Release:           22.04
```

```
Codename:        jammy
```

2) If your computer is running Windows and you don't have Ubuntu 22.04 installed, you can consider using **VirtualBox** or **VMware** to install an Ubuntu 22.04 virtual machine in Windows. But please note that you should not compile orangepi-build on a WSL virtual machine, because orangepi-build has not been tested in a WSL virtual machine, so it is not guaranteed that orangepi-build can be used normally in WSL.

3) The installation image download address of Ubuntu 22.04 **amd64** version is:

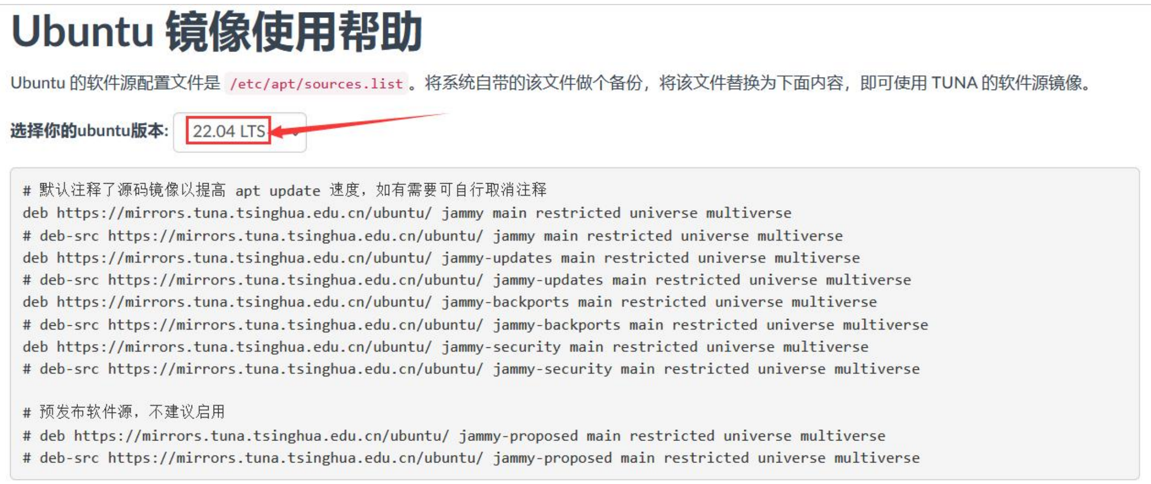
```
https://mirrors.tuna.tsinghua.edu.cn/ubuntu-releases/22.04/ubuntu-22.04.3-desktop-amd64.iso
or
https://repo.huaweicloud.com/ubuntu-releases/22.04/ubuntu-22.04.3-desktop-amd64.iso
```

4) After installing Ubuntu 22.04 on your computer or in a virtual machine, please set the software source of Ubuntu 22.04 to Tsinghua source first, otherwise it is easy to get errors due to network reasons when installing software later

a. For the method of replacing Tsinghua source, please refer to the instructions on this page

```
https://mirrors.tuna.tsinghua.edu.cn/help/ubuntu/
```

b. Note that the Ubuntu version needs to be switched to 22.04



Ubuntu 镜像使用帮助

Ubuntu 的软件源配置文件是 /etc/apt/sources.list。将系统自带的该文件做个备份，将该文件替换为下面内容，即可使用 TUNA 的软件源镜像。

选择你的ubuntu版本: 22.04 LTS

```
# 默认注释了源码镜像以提高 apt update 速度，如有需要可自行取消注释
deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy main restricted universe multiverse
# deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy main restricted universe multiverse
deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-updates main restricted universe multiverse
# deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-updates main restricted universe multiverse
deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-backports main restricted universe multiverse
# deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-backports main restricted universe multiverse
deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-security main restricted universe multiverse
# deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-security main restricted universe multiverse

# 预发布软件源，不建议启用
# deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-proposed main restricted universe multiverse
# deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-proposed main restricted universe multiverse
```

c. The content of the **/etc/apt/sources.list** file that needs to be replaced is

```
test@test:~$ sudo mv /etc/apt/sources.list /etc/apt/sources.list.bak
test@test:~$ sudo vim /etc/apt/sources.list
# The source mirror is commented out by default to increase the speed of apt update. You can uncomment it if necessary.
```

```
deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy main restricted universe multiverse
# deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy main restricted universe multiverse
deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-updates main restricted universe multiverse
# deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-updates main restricted universe multiverse
deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-backports main restricted universe multiverse
# deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-backports main restricted universe multiverse
deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-security main restricted universe multiverse
# deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-security main restricted universe multiverse

# Pre-release software source, not recommended to enable
# deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-proposed main restricted universe multiverse
# deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-proposed main restricted universe multiverse
```

d. After the replacement, you need to update the package information and ensure that there is no error

```
test@test:~$ sudo apt update
```

e. In addition, since the source codes of the kernel and U-boot are stored on GitHub, it is very important to ensure that the computer can download the code from GitHub normally when compiling the image.

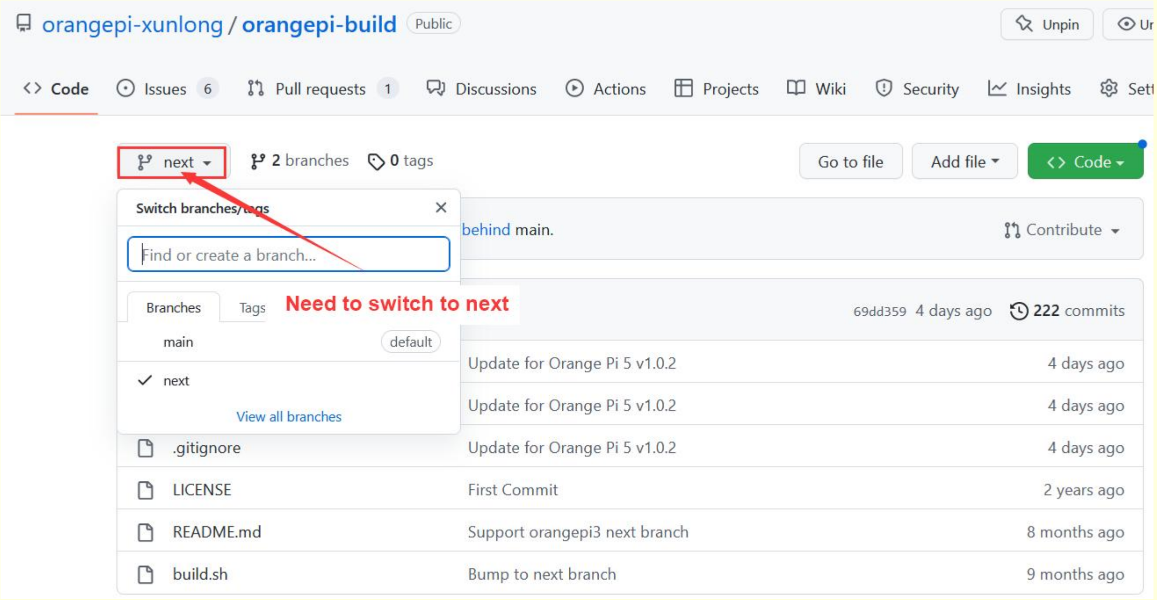
4. 2. Get the source code of Linux SDK

4. 2. 1. Download orangepi-build from github

1) Linux SDK actually refers to the orangepi-build code. Orangepi-build is modified based on the armbian build compilation system. Orangepi-build can be used to compile multiple versions of Linux images. First download the orangepi-build code. The command is as follows:

```
test@test:~$ sudo apt-get update
test@test:~$ sudo apt-get install -y git
test@test:~$ git clone https://github.com/orangepi-xunlong/orangepi-build.git -b next
```

Note that the Orange Pi R2S development board needs to download the next branch source code of orangepi-build. The above git clone command needs to specify the branch of orangepi-build source code as next.



Need to switch to next

behind main.

69dd359 4 days ago 222 commits

	Update for Orange Pi 5 v1.0.2	4 days ago
	Update for Orange Pi 5 v1.0.2	4 days ago
.gitignore	Update for Orange Pi 5 v1.0.2	4 days ago
LICENSE	First Commit	2 years ago
README.md	Support orangepi3 next branch	8 months ago
build.sh	Bump to next branch	9 months ago

You do not need to enter the username and password of your GitHub account when you download the orangepi-build code through the git clone command (the same applies to downloading other codes in this manual). If your Ubuntu PC prompts you to enter the username and password of your GitHub account after entering the git clone command, it is usually because the address of the orangepi-build repository after git clone is entered incorrectly. Please check the command spelling carefully for any errors, instead of thinking that we forgot to provide the username and password of our GitHub account.

2) The u-boot and linux kernel versions currently used by the development board are as follows

Branches	u-boot Version	linuxKernel version
current	u-boot 22.10	Linux6.6

The branch mentioned here is not the same as the branch of orangepi-build source code, please do not confuse them. This branch is mainly used to distinguish different kernel source code versions.

3) After orangepi-build is downloaded, it will contain the following files and folders

a. **build.sh**: Compile the startup script

b. **external**: Contains configuration files, specific scripts, and source code of some programs required for compiling images.

c. **LICENSE**: GPL 2 License File

d. **README.md**:orangepi-build description file

e. **scripts**: Generic script for compiling linux images

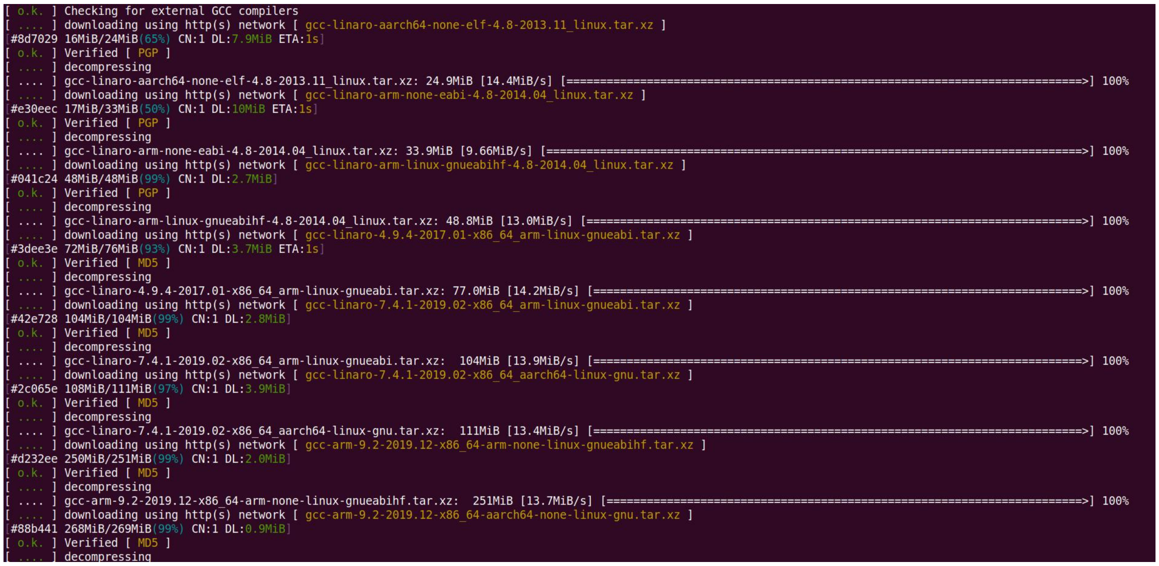
```
test@test:~/orangepi-build$ ls
build.sh    external    LICENSE    README.md    scripts
```

If you download the orangepi-build code from github, you may find that orangepi-build does not contain the source code of u-boot and linux kernel, nor the cross-compilation toolchain required to compile u-boot and linux kernel. This is normal because these things are stored in other separate github repositories or some servers (the addresses will be detailed below). orangepi-build will specify the addresses of u-boot, linux kernel and cross-compilation toolchain in the script and configuration file. When running orangepi-build, if it finds that these things are not available locally, it will automatically download them from the corresponding places.

4. 2. 2. Download the cross-compilation toolchain

The cross-compilation toolchain will only be downloaded when you compile the image using orangepi-build on an x64 computer. Compiling the Linux image of the development board in Ubuntu 22.04 on the development board will not download the cross-compilation toolchain, and orangepi-build/toolchains will be an empty folder.

1) When orangepi-build is run for the first time, it will automatically download the cross-compilation **toolchain** and put it in the toolchains folder. Each time you run the build.sh script of orangepi-build, it will check whether the cross-compilation **toolchain** in toolchains exists. If not, it will restart the download. If it exists, it will be used directly without repeated download.



```
[ o.k. ] Checking for external GCC compilers
[ .... ] downloading using http(s) network [ gcc-linaro-aarch64-none-elf-4.8-2013.11_linux.tar.xz ]
[#8d7029 16MiB/24MiB(65%) CN:1 DL:7.9MiB ETA:1s]
[ o.k. ] Verified [ PGP ]
[ .... ] decompressing
[ .... ] gcc-linaro-aarch64-none-elf-4.8-2013.11_linux.tar.xz: 24.9MiB [14.4MiB/s] [==============================================================================>] 100%
[ .... ] downloading using http(s) network [ gcc-linaro-arm-none-eabi-4.8-2014.04_linux.tar.xz ]
[#e30eec 17MiB/33MiB(50%) CN:1 DL:10MiB ETA:1s]
[ o.k. ] Verified [ PGP ]
[ .... ] decompressing
[ .... ] gcc-linaro-arm-none-eabi-4.8-2014.04_linux.tar.xz: 33.9MiB [9.66MiB/s] [=================================================================================>] 100%
[ .... ] downloading using http(s) network [ gcc-linaro-arm-linux-gnueabihf-4.8-2014.04_linux.tar.xz ]
[#041c24 48MiB/48MiB(99%) CN:1 DL:2.7MiB]
[ o.k. ] Verified [ PGP ]
[ .... ] decompressing
[ .... ] gcc-linaro-arm-linux-gnueabihf-4.8-2014.04_linux.tar.xz: 48.8MiB [13.0MiB/s] [===========================================================================>] 100%
[ .... ] downloading using http(s) network [ gcc-linaro-4.9.4-2017.01-x86_64_arm-linux-gnueabi.tar.xz ]
[#3dee3e 72MiB/76MiB(93%) CN:1 DL:3.7MiB ETA:1s]
[ o.k. ] Verified [ MD5 ]
[ .... ] decompressing
[ .... ] gcc-linaro-4.9.4-2017.01-x86_64_arm-linux-gnueabi.tar.xz: 77.0MiB [14.2MiB/s] [==========================================================================>] 100%
[ .... ] downloading using http(s) network [ gcc-linaro-7.4.1-2019.02-x86_64_arm-linux-gnueabi.tar.xz ]
[#42e728 104MiB/104MiB(99%) CN:1 DL:2.8MiB]
[ o.k. ] Verified [ MD5 ]
[ .... ] decompressing
[ .... ] gcc-linaro-7.4.1-2019.02-x86_64_arm-linux-gnueabi.tar.xz:  104MiB [13.9MiB/s] [==========================================================================>] 100%
[ .... ] downloading using http(s) network [ gcc-linaro-7.4.1-2019.02-x86_64_aarch64-linux-gnu.tar.xz ]
[#2c065e 108MiB/111MiB(97%) CN:1 DL:3.9MiB]
[ o.k. ] Verified [ MD5 ]
[ .... ] decompressing
[ .... ] gcc-linaro-7.4.1-2019.02-x86_64_aarch64-linux-gnu.tar.xz:  111MiB [13.4MiB/s] [==========================================================================>] 100%
[ .... ] downloading using http(s) network [ gcc-arm-9.2-2019.12-x86_64-arm-none-linux-gnueabihf.tar.xz ]
[#d232ee 250MiB/251MiB(99%) CN:1 DL:2.0MiB]
[ o.k. ] Verified [ MD5 ]
[ .... ] decompressing
[ .... ] gcc-arm-9.2-2019.12-x86_64-arm-none-linux-gnueabihf.tar.xz:  251MiB [13.7MiB/s] [========================================================================>] 100%
[ .... ] downloading using http(s) network [ gcc-arm-9.2-2019.12-x86_64-aarch64-none-linux-gnu.tar.xz ]
[#88b441 268MiB/269MiB(99%) CN:1 DL:0.9MiB]
[ o.k. ] Verified [ MD5 ]
[ .... ] decompressing
```

2) The mirror website of the cross-compilation tool chain in China is the open source software mirror website of Tsinghua University

```
https://mirrors.tuna.tsinghua.edu.cn/armbian-releases/_toolchain/
```

3) After downloading **toolchains**, it will contain multiple versions of cross-compilation toolchains, and the development board will only use two of them

```
test@test:~/orangepi-build$ ls toolchains/
gcc-arm-11.2-2022.02-x86_64-aarch64-none-linux-gnu
gcc-arm-11.2-2022.02-x86_64-arm-none-linux-gnueabihf
gcc-arm-9.2-2019.12-x86_64-aarch64-none-linux-gnu
gcc-arm-9.2-2019.12-x86_64-arm-none-linux-gnueabihf
gcc-linaro-4.9.4-2017.01-x86_64_arm-linux-gnueabi
gcc-linaro-5.5.0-2017.10-x86_64_arm-linux-gnueabihf
gcc-linaro-7.4.1-2019.02-x86_64_aarch64-linux-gnu
gcc-linaro-7.4.1-2019.02-x86_64_arm-linux-gnueabi
gcc-linaro-aarch64-none-elf-4.8-2013.11_linux
gcc-linaro-arm-linux-gnueabihf-4.8-2014.04_linux
gcc-linaro-arm-none-eabi-4.8-2014.04_linux
```

4) The cross-compilation tool chain used to compile the Linux kernel source code is

a. Linux6.6

```
riscv64-unknown-linux-gnu-gcc
```

5) The cross-compilation tool chain used to compile the u-boot source code is

a. v2022.10

```
riscv64-unknown-linux-gnu-gcc
```

4. 2. 3. orangepi-build complete directory structure description

1) After downloading the orangepi-build repository, it does not contain the source code of the Linux kernel, u-boot, and the cross-compilation toolchain. The source code of the Linux kernel and u-boot is stored in a separate git repository.

a. The git repository where the Linux kernel source code is stored is as follows：

```
https://github.com/orangepi-xunlong/linux-orangepi/tree/orange-pi-6.6-ky
```

b. The git repository where the u-boot source code is stored is as follows:

```
https://github.com/orangepi-xunlong/u-boot-orangepi/tree/v2022.10-ky
```

2) When orangepi-build is run for the first time, it will download the cross-compilation toolchain, u-boot and Linux kernel source code. After successfully compiling a Linux image, the files and folders that can be seen in orangepi-build are

a. **build.sh**: Compile the startup script

b. **external**: Contains configuration files, scripts for specific functions, and source code for some programs needed to compile the image. The rootfs compressed package cached during the image compilation process is also stored in external

c. **kernel**: The source code of the Linux kernel is stored in the folder named **orange-pi-6.6-ky**, which is the kernel source code of the current branch of the Orange Pi R2S development board. Please do not manually modify the name of the kernel source code folder. If modified, the kernel source code will be downloaded again when the compile system is running.

d. **LICENSE**: GPL 2 License File

e. **README.md**: orangepi-build documentation

f. **output**: Store the compiled u-boot, linux and other deb packages, compilation logs, compiled images and other files

g. **scripts**: Generic script for compiling linux images

h. **toolchains**: Store cross-compilation toolchain

i. **u-boot**: The u-boot source code is stored in the folder named **v2022.10-ky**, which stores the u-boot source code of the current branch of the Orange Pi R2S series development board. Please do not manually modify the name of the u-boot source code folder. If modified, the u-boot source code will be re-downloaded when the compilation system is running.

j. **userpatches**: Store the configuration files needed to compile the script

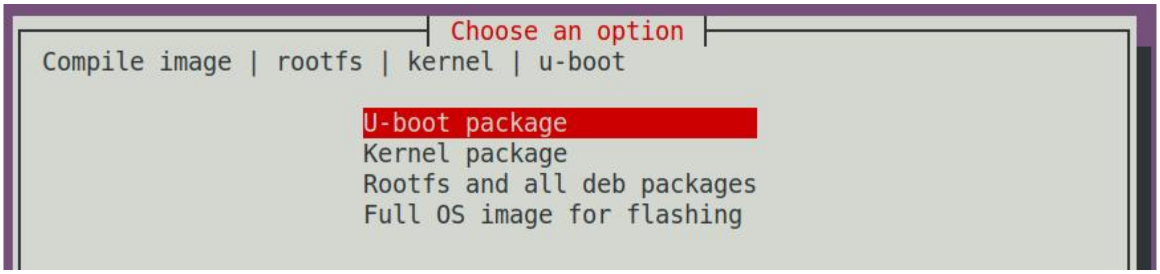
```
test@test:~/orangepi-build$ ls
build.sh    external    kernel    LICENSE    output    README.md    scripts
toolchains   u-boot   userpatches
```

4. 3. Compile u-boot

1) Run the build.sh script, remember to add sudo permissions

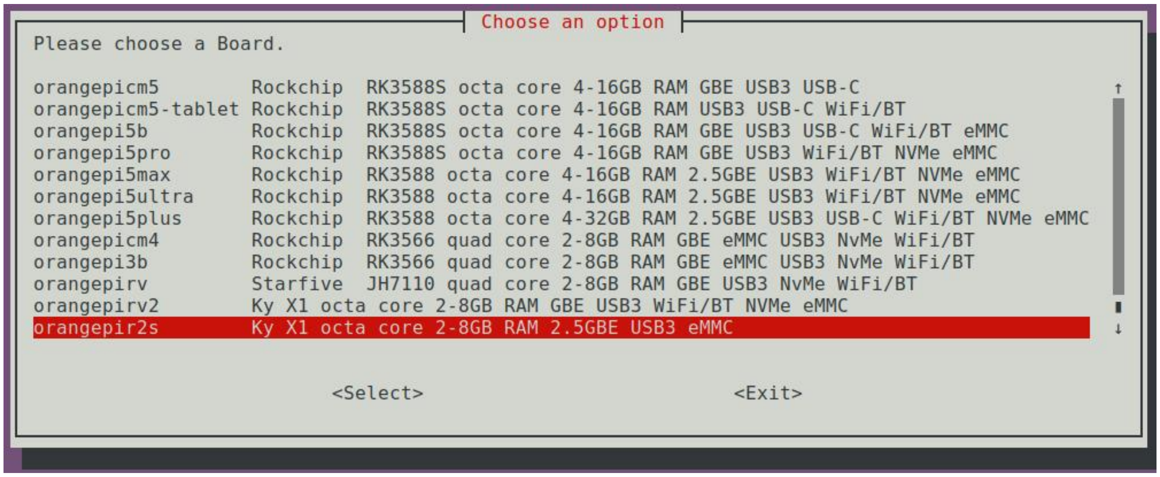
```
test@test:~/orangepi-build$ sudo ./build.sh
```

2) Select **U-boot package** and press Enter



```
Choose an option
Compile image | rootfs | kernel | u-boot

U-boot package
Kernel package
Rootfs and all deb packages
Full OS image for flashing
```

3) Then select the model of the development board



```
Choose an option
Please choose a Board.

orangepicm5          Rockchip  RK3588S octa core 4-16GB RAM GBE USB3 USB-C
orangepicm5-tablet   Rockchip  RK3588S octa core 4-16GB RAM USB3 USB-C WiFi/BT
orangepi5b           Rockchip  RK3588S octa core 4-16GB RAM GBE USB3 USB-C WiFi/BT eMMC
orangepi5pro         Rockchip  RK3588S octa core 4-16GB RAM GBE USB3 WiFi/BT NVMe eMMC
orangepi5max         Rockchip  RK3588 octa core 4-16GB RAM 2.5GBE USB3 WiFi/BT NVMe eMMC
orangepi5ultra       Rockchip  RK3588 octa core 4-16GB RAM 2.5GBE USB3 WiFi/BT NVMe eMMC
orangepi5plus        Rockchip  RK3588 octa core 4-32GB RAM 2.5GBE USB3 USB-C WiFi/BT NVMe eMMC
orangepicm4          Rockchip  RK3566 quad core 2-8GB RAM GBE eMMC USB3 NvMe WiFi/BT
orangepi3b           Rockchip  RK3566 quad core 2-8GB RAM GBE eMMC USB3 NvMe WiFi/BT
orangepirv           Starfive  JH7110 quad core 2-8GB RAM GBE USB3 NvMe WiFi/BT
orangepirv2          Ky X1 octa core 2-8GB RAM GBE USB3 WiFi/BT NVMe eMMC
orangepir2s          Ky X1 octa core 2-8GB RAM 2.5GBE USB3 eMMC

              <Select>                          <Exit>
```

4) Then u-boot will start to compile. Some of the information prompted during compilation is as follows

a. u-boot source code version

```
[ o.k. ] Compiling u-boot [ v2022.10 ]
```

b. Version of the cross-compilation toolchain

```
[ o.k. ] Compiler version [ riscv64-unknown-linux-gnu-gcc 13.2.1 ]
```

c. The path of the compiled u-boot deb package

```
[ o.k. ] Target directory [ orangepi-build/output/debs/u-boot ]
```

d. The package name of the compiled u-boot deb package

```
[ o.k. ] File name [ linux-u-boot-current-orangepir2s_1.0.0_riscv64.deb ]
```

e. Compilation time

```
[ o.k. ] Runtime [ 1 min ]
```

f. Repeat the command to compile u-boot. Use the following command to start compiling u-boot directly without selecting through the graphical interface.

```
[ o.k. ] Repeat Build Options [ sudo ./build.sh BOARD=orangepir2s
BRANCH=current BUILD_OPT=u-boot KERNEL_CONFIGURE=no ]
```

5) View the compiled u-boot deb package

```
test@test:~/orangepi-build$ ls output/debs/u-boot/
linux-u-boot-current-orangepir2s_1.0.0_riscv64.deb
```

6) The files contained in the generated u-boot deb package are as follows

a. Use the following command to decompress the deb package

```
test@test:~/orangepi-build$ cd output/debs/u-boot
test@test:~/orangepi_build/output/debs/u-boot$ $ dpkg -x \
linux-u-boot-current-orangepir2s_1.0.0_riscv64.deb .    (Note that there is a “.”)
test@test:~/orangepi_build/output/debs/u-boot$ ls
linux-u-boot-current-orangepir2s_1.0.0_riscv64.deb    usr
```

b. The decompressed files are as follows

```
test@test:~/orangepi-build/output/debs/u-boot$ tree usr
usr
└── lib
    ├── linux-u-boot-current-orangepir2s_1.0.0_riscv64
    │   ├── bootinfo_emmc.bin
    │   ├── bootinfo_sd.bin
    │   ├── bootinfo_spinor.bin
    │   ├── FSBL.bin
    │   ├── u-boot-env-default.bin
    │     └── u-boot-opensbi.itb
    └── u-boot
          ├── LICENSE
          ├── platform_install.sh
          └── x1_defconfig

3 directories, 9 files
```

7) When the orangepi-bulid compilation system compiles the u-boot source code, it will first synchronize the u-boot source code with the u-boot source code on the github server. So if you want to modify the u-boot source code, you first need to turn off the source code download and update function (**you need to fully compile u-boot once before turning off this function, otherwise it will prompt that the u-boot source code cannot be found. If the source code compression package is downloaded from Baidu Cloud Disk, there will be no such problem, because the u-boot source code has been cached**), otherwise the changes made will be restored. The method is as follows:

Set the IGNORE_UPDATES variable in userpatches/config-default.conf to "yes"

```
test@test:~/orangepi-build$ vim userpatches/config-default.conf
IGNORE_UPDATES="yes"
```

8) When debugging the u-boot code, you can use the following method to update the u-boot in the Linux image for testing

a. Upload the compiled u-boot deb package to the Linux system of the development board

```
test@test:~/orangepi-build$ cd output/debs/u-boot
test@test:~/orangepi_build/output/debs/u-boot$ scp \
linux-u-boot-current-orangepir2s_1.0.0_riscv64.deb root@192.168.1.xxx:/root
```

b. Then log in to the development board and uninstall the installed u-boot deb package

```
root@orangepi:~# apt purge -y linux-u-boot-orangepir2s-current
```

c. Install the new u-boot deb package just uploaded

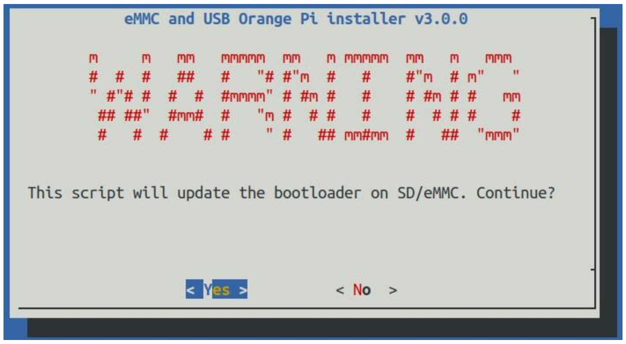
```
root@orangepi:~# dpkg -i linux-u-boot-current-orangepir2s_1.0.0_riscv64.deb
```

d. Then run the nand-sata-install script

```
root@orangepi:~# nand-sata-install
```

e. Then select **5 Install/Update the bootloader on SD/eMM** to update the u-boot in the eMMC card

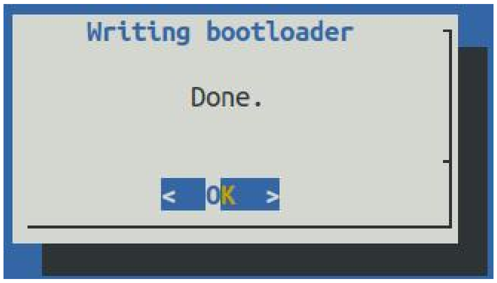
f. After pressing the Enter key, a Warning will pop up first.



eMMC and USB Orange Pi installer v3.0.0

This script will update the bootloader on SD/eMMC. Continue?

g. Press the Enter key again to start updating u-boot. After the update, the following information will be displayed



Writing bootloader

Done.

h. Then you can restart the development board to test whether the changes to u-boot are effective.

9) Other useful information

a. In the u-boot 2022.10 source code, the defconfig configuration file used by the development board is

```
orangepi-build/u-boot/v2022.10-ky/configs/x1_defconfig
```

b. In the u-boot 2022.10 source code, the development board uses the dts file as

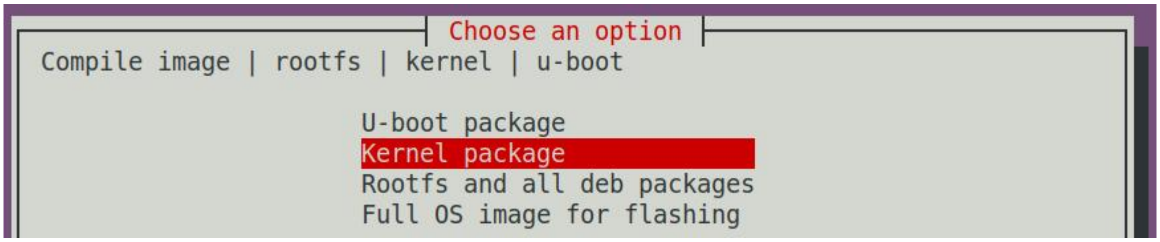
```
orangepi-build/u-boot/v2022.10-ky/arch/riscv/dts/x1_orangepi-r2s.dts
```

4. 4. Compile the Linux kernel

1) Run the build.sh script and remember to add sudo permissions

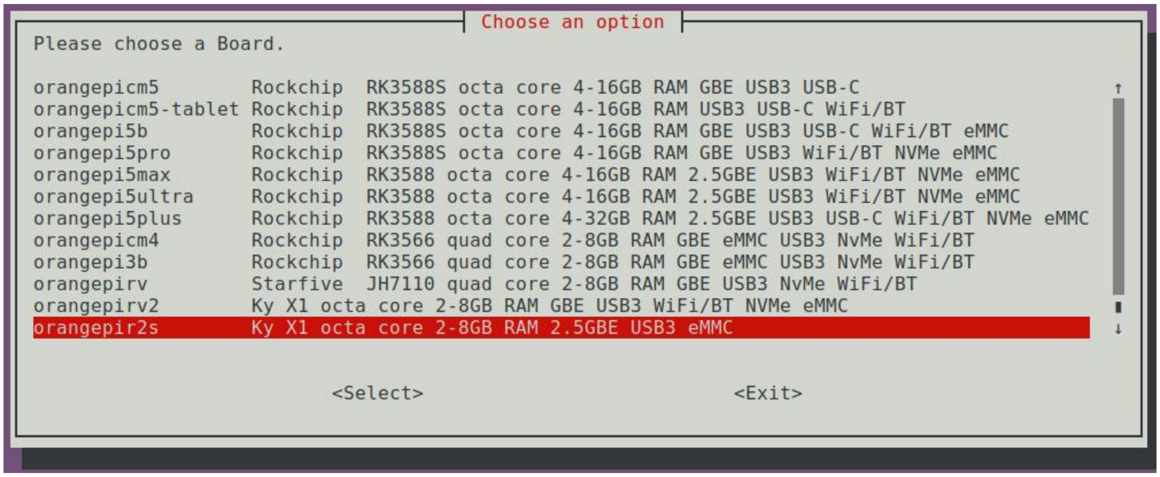
```
test@test:~/orangepi-build$ sudo ./build.sh
```

2) Select **Kernel package** and press Enter



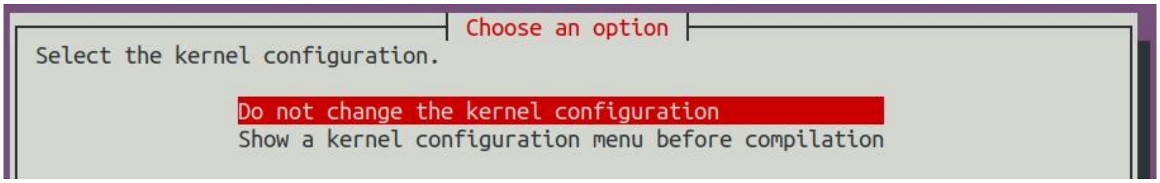
```
Choose an option
Compile image | rootfs | kernel | u-boot

                    U-boot package
                    Kernel package
                    Rootfs and all deb packages
                    Full OS image for flashing
```

3) Then select the model of the development board



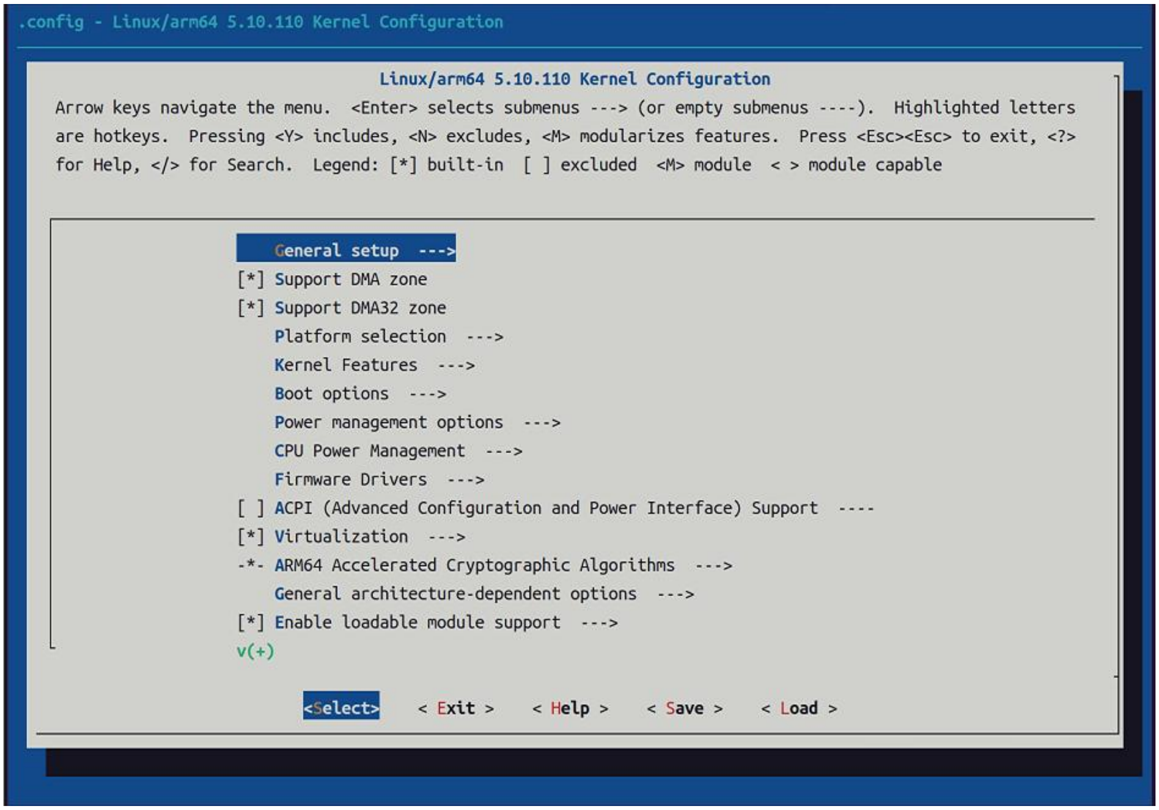
```
Choose an option
Please choose a Board.

orangepicm5        Rockchip  RK3588S octa core 4-16GB RAM GBE USB3 USB-C
orangepicm5-tablet Rockchip  RK3588S octa core 4-16GB RAM USB3 USB-C WiFi/BT
orangepi5b         Rockchip  RK3588S octa core 4-16GB RAM GBE USB3 USB-C WiFi/BT eMMC
orangepi5pro       Rockchip  RK3588S octa core 4-16GB RAM GBE USB3 WiFi/BT NVMe eMMC
orangepi5max       Rockchip  RK3588 octa core 4-16GB RAM 2.5GBE USB3 WiFi/BT NVMe eMMC
orangepi5ultra     Rockchip  RK3588 octa core 4-16GB RAM 2.5GBE USB3 WiFi/BT NVMe eMMC
orangepi5plus      Rockchip  RK3588 octa core 4-32GB RAM 2.5GBE USB3 USB-C WiFi/BT NVMe eMMC
orangepicm4        Rockchip  RK3566 quad core 2-8GB RAM GBE eMMC USB3 NvMe WiFi/BT
orangepi3b         Rockchip  RK3566 quad core 2-8GB RAM GBE eMMC USB3 NvMe WiFi/BT
orangepirv         Starfive  JH7110 quad core 2-8GB RAM GBE USB3 NvMe WiFi/BT
orangepirv2        Ky X1 octa core 2-8GB RAM GBE USB3 WiFi/BT NVMe eMMC
orangepir2s        Ky X1 octa core 2-8GB RAM 2.5GBE USB3 eMMC

                    <Select>                          <Exit>
```

4) Then you will be prompted whether you need to display the kernel configuration interface. If you do not need to modify the kernel configuration, select the first one. If you need to modify the kernel configuration, select the second one.



```
Choose an option
Select the kernel configuration.

            Do not change the kernel configuration
            Show a kernel configuration menu before compilation
```

5) If you selected the option to display the kernel configuration menu (the second option) in step 4), the kernel configuration interface opened by **make menuconfig** will pop up. You can modify the kernel configuration directly at this time. After modifying, save and exit. After exiting, the kernel source code will be compiled.



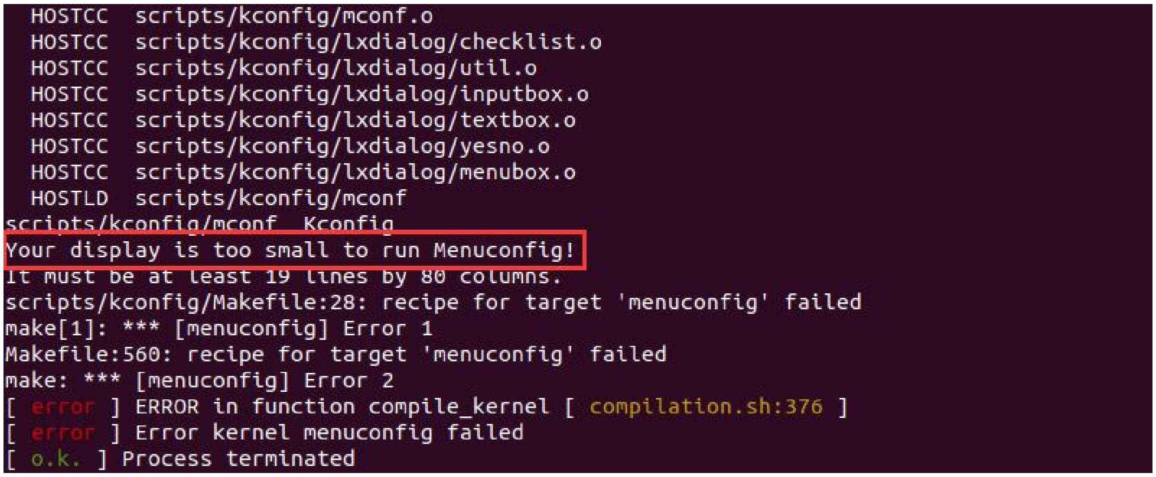
```
.config - Linux/arm64 5.10.110 Kernel Configuration

                         Linux/arm64 5.10.110 Kernel Configuration
  Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters
  are hotkeys.  Pressing <Y> includes, <N> excludes, <M> modularizes features.  Press <Esc><Esc> to exit, <?>
  for Help, </> for Search.  Legend: [*] built-in  [ ] excluded  <M> module  < > module capable

                    General setup  --->
                [*] Support DMA zone
                [*] Support DMA32 zone
                    Platform selection  --->
                    Kernel Features  --->
                    Boot options  --->
                    Power management options  --->
                    CPU Power Management  --->
                    Firmware Drivers  --->
                [ ] ACPI (Advanced Configuration and Power Interface) Support  ----
                [*] Virtualization  --->
                -*- ARM64 Accelerated Cryptographic Algorithms  --->
                    General architecture-dependent options  --->
                [*] Enable loadable module support  --->
                v(+)

                      <Select>    < Exit >    < Help >    < Save >    < Load >
```

a. If you do not need to modify the kernel configuration options, when running the build.sh script, pass in **KERNEL_CONFIGURE=no** to temporarily block the kernel configuration interface from popping up.

```
test@test:~/orangepi-build$ sudo ./build.sh KERNEL_CONFIGURE=no
```

b. You can also set **KERNEL_CONFIGURE=no** in the **orangepi-build/userpatches/config-default.conf** configuration file to permanently disable this feature

c. If the following error message appears when compiling the kernel, it is because the terminal interface of the Ubuntu PC is too small, which causes the **make menuconfig** interface to fail to display. Please adjust the terminal of the Ubuntu PC to the maximum size and re-run the build.sh script.



```
  HOSTCC  scripts/kconfig/mconf.o
  HOSTCC  scripts/kconfig/lxdialog/checklist.o
  HOSTCC  scripts/kconfig/lxdialog/util.o
  HOSTCC  scripts/kconfig/lxdialog/inputbox.o
  HOSTCC  scripts/kconfig/lxdialog/textbox.o
  HOSTCC  scripts/kconfig/lxdialog/yesno.o
  HOSTCC  scripts/kconfig/lxdialog/menubox.o
  HOSTLD  scripts/kconfig/mconf
scripts/kconfig/mconf  Kconfig
Your display is too small to run Menuconfig!
It must be at least 19 lines by 80 columns.
scripts/kconfig/Makefile:28: recipe for target 'menuconfig' failed
make[1]: *** [menuconfig] Error 1
Makefile:560: recipe for target 'menuconfig' failed
make: *** [menuconfig] Error 2
[ error ] ERROR in function compile_kernel [ compilation.sh:376 ]
[ error ] Error kernel menuconfig failed
[ o.k. ] Process terminated
```

6) Some of the information prompted when compiling the kernel source code is as follows

a. Linux kernel source code version

[o.k.] Compiling current kernel [**6.6.63**]

b. Version of the cross-compilation toolchain used

[o.k.] Compiler version [**riscv64-unknown-linux-gnu-gcc 13.2.1**]

c. The default configuration file used by the kernel and the path where it is stored

[o.k.] Using kernel config file [**config/kernel/linux-ky-current.config**]

d. The path of the compiled kernel-related deb package

[o.k.] Target directory [**orangepi-build/output/debs/**]

e. The package name of the compiled kernel image deb package

[o.k.] File name [**linux-image-current-ky_1.0.0_riscv64.deb**]

f. Compilation time

[o.k.] Runtime [**5 min**]

g. Finally, the compilation command for the last selected kernel will be displayed. Use the following command to directly start compiling the kernel source code without selecting through the graphical interface.

[o.k.] Repeat Build Options [**sudo ./build.sh BOARD=orangepir2s BRANCH=current BUILD_OPT=kernel KERNEL_CONFIGURE=no**]

7) Check the compiled kernel-related deb packages

a. **linux-dtb-current-ky_1.0.0_riscv64.deb** Contains dtb files used by the kernel

b. **linux-headers-current-ky_1.0.0_riscv64.deb** Include kernel header files

c. **linux-image-current-ky_1.0.0_riscv64.deb** Contains kernel images and kernel modules

```
test@test:~/orangepi-build$ ls output/debs/linux-*
output/debs/linux-dtb-current-ky_1.0.0_riscv64.deb
output/debs/linux-image-current-ky_1.0.0_riscv64.deb
output/debs/linux-headers-current-ky_1.0.0_riscv64.deb
```

8) The generated linux-image deb package contains the following files

a. Use the following command to unzip the deb package

```
test@test:~/orangepi-build$ cd output/debs
test@test:~/orangepi_build/output/debs$ mkdir test
test@test:~/orangepi_build/output/debs$ cp \
linux-image-current-ky_1.0.0_riscv64.deb test/
test@test:~/orangepi_build/output/debs$ cd test
test@test:~/orangepi_build/output/debs/test$ dpkg -x \
linux-image-current-ky_1.0.0_riscv64.deb .
test@test:~/orangepi_build/output/debs/test$ ls
boot   etc   lib   linux-image-current-ky_1.0.0_riscv64.deb   usr
```

b. The decompressed files are as follows

```
test@test:~/orangepi-build/output/debs/test$ tree -L 2
.
├── boot
│   ├── config-6.6.63-ky
│   ├── System.map-6.6.63-ky
│   └── vmlinuz-6.6.63-ky
├── etc
│   └── kernel
├── lib
│   └── modules
├── linux-image-current-ky_1.0.0_riscv64.deb
└── usr
    ├── lib
    └── share
```

9) When the orangepi-bulid compilation system compiles the Linux kernel source code, it will first synchronize the Linux kernel source code with the Linux kernel source code on the GitHub server. So if you want to modify the Linux kernel source code, you first

need to turn off the source code update function (**you need to fully compile the Linux kernel source code once before turning off this function, otherwise it will prompt that the Linux kernel source code cannot be found. If the source code compression package is downloaded from Baidu Cloud Disk, there will be no such problem because the Linux source code has been cached**), otherwise the changes made will be restored. The method is as follows:

Set the IGNORE_UPDATES variable in **userpatches/config-default.conf to** "yes"

```
test@test:~/orangepi-build$ vim userpatches/config-default.conf
IGNORE_UPDATES="yes"
```

10) If the kernel is modified, you can use the following method to update the kernel and kernel modules of the development board Linux system

a. Upload the compiled Linux kernel deb package to the Linux system of the development board

```
test@test:~/orangepi-build$ cd output/debs
test@test:~/orangepi-build/output/debs$ scp \
linux-image-current-ky_1.0.0_riscv64.deb root@192.168.1.xxx:/root
```

b. Then log in to the development board and uninstall the installed linux kernel deb package

```
root@orangepi:~# apt purge -y linux-image-current-ky
```

c. Then install the new Linux kernel deb package just uploaded

```
root@orangepi:~# dpkg -i linux-image-current-ky_1.0.0_riscv64.deb
```

d. Then restart the development board and check whether the kernel-related modifications have taken effect.

```
root@orangepi:~# reboot
```

10) Other useful information

a. The kernel configuration file is stored in the following location. Please do not look for the kernel configuration file used by the development board in the kernel source code.

```
orangepi-build/external/config/kernel/linux-ky-current.config
```

b. The location of the dts file used by the development board is

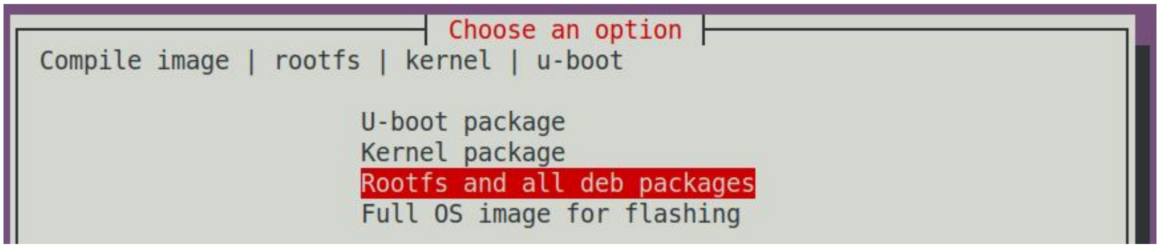
```
orangepi-build/kernel/orange-pi-6.6-ky/arch/riscv/boot/dts/ky/x1_orangepi-r2s.dts
```

4. 5. Compile rootfs

1) Run the build.sh script and remember to add sudo permissions

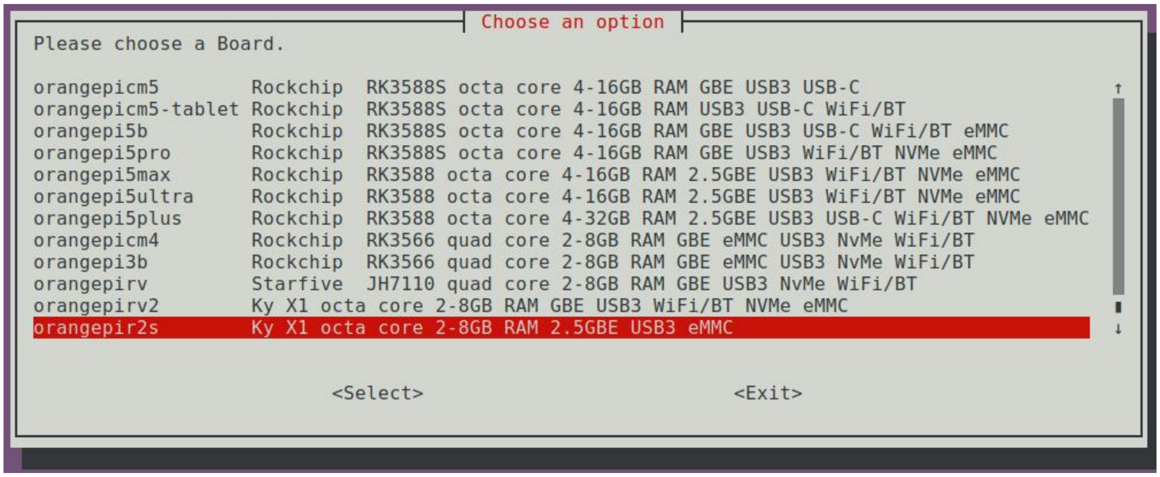
```
test@test:~/orangepi-build$ sudo ./build.sh
```

2) Select **Rootfs and all deb packages** and press Enter



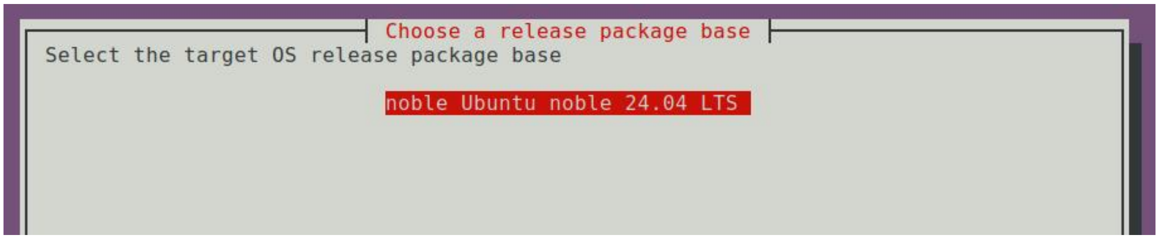
```
Choose an option
Compile image | rootfs | kernel | u-boot

            U-boot package
            Kernel package
            Rootfs and all deb packages
            Full OS image for flashing
```

3) Then select the model of the development board



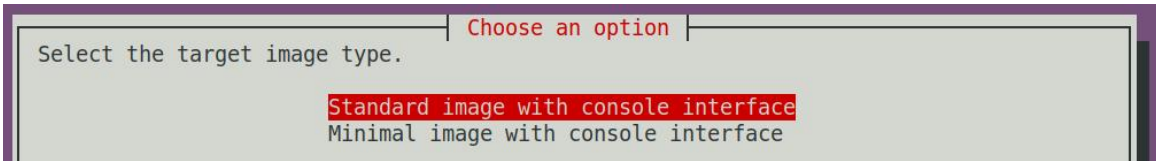
```
Choose an option
Please choose a Board.

orangepicm5         Rockchip  RK3588S octa core 4-16GB RAM GBE USB3 USB-C
orangepicm5-tablet  Rockchip  RK3588S octa core 4-16GB RAM USB3 USB-C WiFi/BT
orangepi5b          Rockchip  RK3588S octa core 4-16GB RAM GBE USB3 USB-C WiFi/BT eMMC
orangepi5pro        Rockchip  RK3588S octa core 4-16GB RAM GBE USB3 WiFi/BT NVMe eMMC
orangepi5max        Rockchip  RK3588 octa core 4-16GB RAM 2.5GBE USB3 WiFi/BT NVMe eMMC
orangepi5ultra      Rockchip  RK3588 octa core 4-16GB RAM 2.5GBE USB3 WiFi/BT NVMe eMMC
orangepi5plus       Rockchip  RK3588 octa core 4-32GB RAM 2.5GBE USB3 USB-C WiFi/BT NVMe eMMC
orangepicm4         Rockchip  RK3566 quad core 2-8GB RAM GBE eMMC USB3 NvMe WiFi/BT
orangepi3b          Rockchip  RK3566 quad core 2-8GB RAM GBE eMMC USB3 NvMe WiFi/BT
orangepirv          Starfive  JH7110 quad core 2-8GB RAM GBE USB3 NvMe WiFi/BT
orangepirv2         Ky X1 octa core 2-8GB RAM GBE USB3 WiFi/BT NVMe eMMC
orangepir2s         Ky X1 octa core 2-8GB RAM 2.5GBE USB3 eMMC

                <Select>                        <Exit>
```

4) Then select the type of rootfs



```
Choose a release package base
Select the target OS release package base

            noble Ubuntu noble 24.04 LTS
```

5) Then choose to compile the Standard version or Minimal version. The Minimal version will have much less pre-installed software than the Standard version (**please do not choose the Minimal version if you do not have special needs, because many things are not pre-installed by default and some functions may not be used**)



```
Choose an option
Select the target image type.

Standard image with console interface
Minimal image with console interface
```

6) Then it will start compiling rootfs. Some of the information prompted during compilation is as follows

a. Type of rootfs

```
[ o.k. ] local not found [ Creating new rootfs cache for noble]
```

b. Storage path of the compiled rootfs compressed package

```
[ o.k. ] Target directory [ external/cache/rootfs ]
```

c. The name of the rootfs compressed package generated by compilation

```
[ o.k. ] File name [ noble-cli-riscv64.ef7fa533e64f5a838939560d81632155.tar.lz4 ]
```

d. Compilation time

```
[ o.k. ] Runtime [ 13 min ]
```

7) View the compiled rootfs compressed package

a. **noble-cli-riscv64.ef7fa533e64f5a838939560d81632155.tar.lz4** is the compressed package of rootfs. The meaning of each field of the name is

a) **Noble** indicates the type of Linux distribution of rootfs

b) **cli** indicates that the rootfs is of server version type. If it is **cli**, it indicates server version type

c) **riscv64** indicates the architecture type of rootfs

d) **ef7fa533e64f5a838939560d81632155** is the MD5 hash value generated by the package names of all packages installed by rootfs. As long as the list of packages installed by rootfs is not modified, this value will not change. The compilation script will use this MD5 hash value to determine whether the rootfs needs to be recompiled.

b. **noble-cli-riscv64.ef7fa533e64f5a838939560d81632155.tar.lz4.list** lists the package names of all software packages installed by rootfs

```
test@test:~/orangepi-build$ ls external/cache/rootfs/
noble-cli-riscv64.ef7fa533e64f5a838939560d81632155.tar.lz4
noble-cli-riscv64.ef7fa533e64f5a838939560d81632155.tar.lz4.current
noble-cli-riscv64.ef7fa533e64f5a838939560d81632155.tar.lz4.list
```

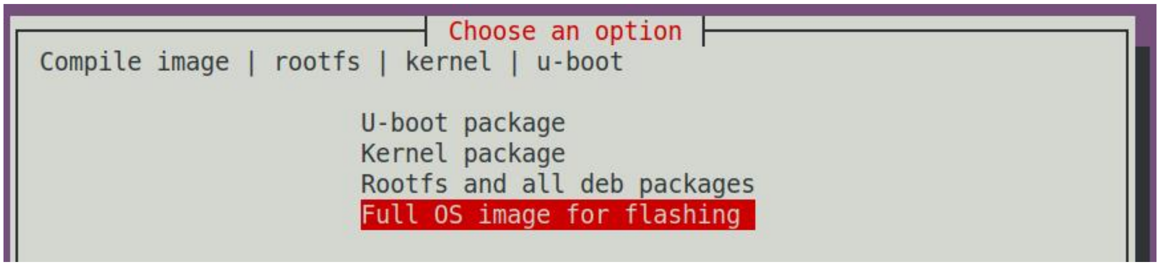
8) If the required rootfs already exists in **external/cache/rootfs**, then compiling rootfs again will skip the compilation process directly and will not restart the compilation. When compiling the image, it will also check whether there is a cached rootfs in **external/cache/rootfs**. If there is, it will be used directly, which can save a lot of download and compilation time.

4. 6. Compile Linux image

1) Run the build.sh script and remember to add sudo permissions

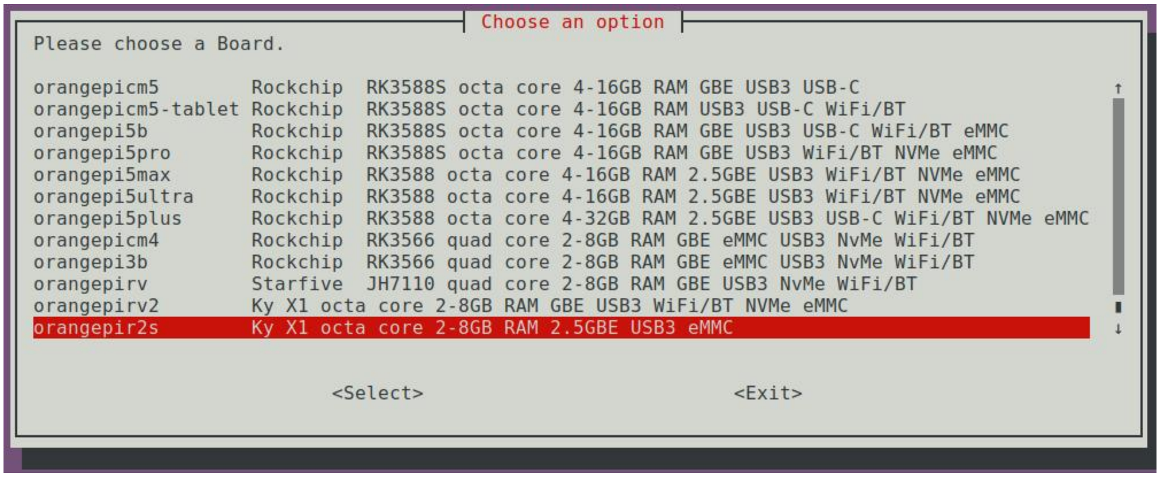
```
test@test:~/orangepi-build$ sudo ./build.sh
```

2) Select **Full OS image for flashing** and press Enter



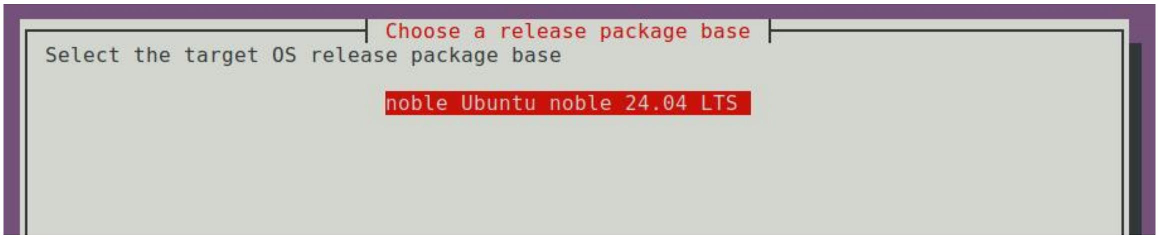
```
Choose an option
Compile image | rootfs | kernel | u-boot

                    U-boot package
                    Kernel package
                    Rootfs and all deb packages
                    Full OS image for flashing
```

3) Then select the model of the development board



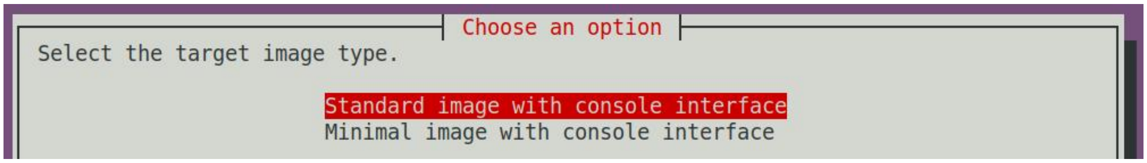
```
Choose an option
Please choose a Board.

orangepicm5          Rockchip  RK3588S octa core 4-16GB RAM GBE USB3 USB-C
orangepicm5-tablet   Rockchip  RK3588S octa core 4-16GB RAM USB3 USB-C WiFi/BT
orangepi5b           Rockchip  RK3588S octa core 4-16GB RAM GBE USB3 USB-C WiFi/BT eMMC
orangepi5pro         Rockchip  RK3588S octa core 4-16GB RAM GBE USB3 WiFi/BT NVMe eMMC
orangepi5max         Rockchip  RK3588 octa core 4-16GB RAM 2.5GBE USB3 WiFi/BT NVMe eMMC
orangepi5ultra       Rockchip  RK3588 octa core 4-16GB RAM 2.5GBE USB3 WiFi/BT NVMe eMMC
orangepi5plus        Rockchip  RK3588 octa core 4-32GB RAM 2.5GBE USB3 USB-C WiFi/BT NVMe eMMC
orangepicm4          Rockchip  RK3566 quad core 2-8GB RAM GBE eMMC USB3 NvMe WiFi/BT
orangepi3b           Rockchip  RK3566 quad core 2-8GB RAM GBE eMMC USB3 NvMe WiFi/BT
orangepirv           Starfive  JH7110 quad core 2-8GB RAM GBE USB3 NvMe WiFi/BT
orangepirv2          Ky X1 octa core 2-8GB RAM GBE USB3 WiFi/BT NVMe eMMC
orangepir2s          Ky X1 octa core 2-8GB RAM 2.5GBE USB3 eMMC

                 <Select>                          <Exit>
```

4) Then select the type of rootfs



```
Choose a release package base
Select the target OS release package base

noble Ubuntu noble 24.04 LTS
```

5) Then choose to compile the Standard version or Minimal version. The Minimal version will have much less pre-installed software than the Standard version (**please do not choose the Minimal version if you do not have special needs, because many things are not pre-installed by default and some functions may not be used**)



```
Choose an option
Select the target image type.

Standard image with console interface
Minimal image with console interface
```

6) Then the Linux image will be compiled. The general process of compilation is as follows

a. Initialize the compilation environment of Ubuntu PC and install the software packages required for the compilation process
b. Download the source code of u-boot and linux kernel (if already cached, just update the code)
c. Compile the u-boot source code and generate the deb package of u-boot
d. Compile the Linux source code and generate Linux-related deb packages
e. Make a deb package for Linux firmware
f. Make the deb package of orangepi-config tool
g. Make a deb package for board-level support
h. If you compile a desktop version image, you will also create a desktop-related deb package
i. Check if the rootfs is already cached. If not, re-create the rootfs. If it is already cached, decompress it and use it directly.
j. Install the deb package generated previously to rootfs
k. Make some specific settings for different development boards and different types of images, such as pre-installing additional software packages, modifying system configuration, etc.
l. Then create an image file and format the partition. The default type is ext4
m. Then copy the configured rootfs to the mirrored partition

n. Then update initramfs

o. Finally, write the u-boot bin file into the image using the dd command

7) After compiling the image, the following information will be prompted

a. Storage path of the compiled image

```
[ o.k. ] Done building
[ output/images/Orangepir2s_1.0.0_ubuntu_noble_server_linux6.6.63/Orangepir2s_
1.0.0_ubuntu_noble_server_linux6.6.63.img ]
```

b. Compilation time

```
[ o.k. ] Runtime [ 19 min ]
```

a. Repeat the command to compile the image. Use the following command to start compiling the image directly without selecting through the graphical interface.

```
[ o.k. ] Repeat Build Options [ sudo ./build.sh    BOARD=orangepir2s
BRANCH=current BUILD_OPT=image RELEASE=noble BUILD_MINIMAL=no
BUILD_DESKTOP=no KERNEL_CONFIGURE=yes ]
```

5. OpenWRT system instructions

5. 1. OpenWRT version

OpenWRT Version	Kernel version
v24.10.0	Linux6.6.73

5. 2. OpenWRT Adaptation

Function	OpenWRT
USB2.0x1	OK
USB3.0x1	OK
3pin Debug serial port	OK
2.5G PCIe Network port*2	OK
Gigabit Ethernet*2	OK
Network port status light	OK
RTL8821CU USB Network Card	OK
RTL8723BU USB Network Card	OK
eMMC	OK

5. 3. Expand rootfs at the first startup

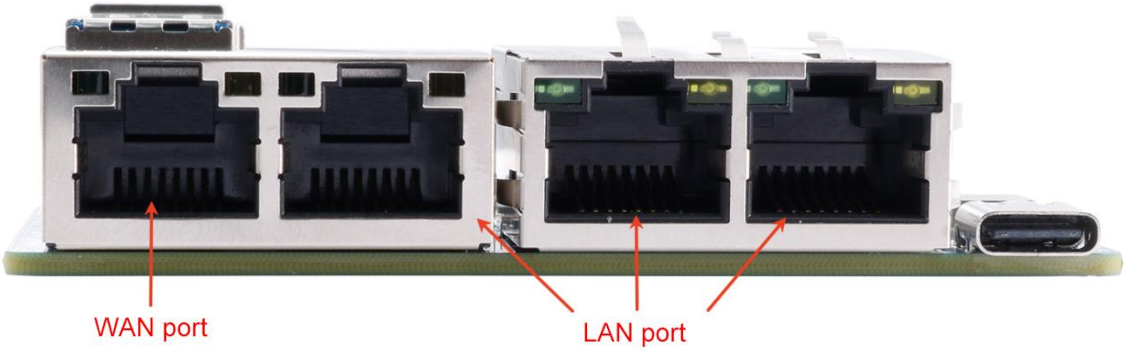
1) When you start the OpenWRT system for the first time, the **resize-rootfs.sh** script will be executed to expand the rootfs, and it will automatically restart after the expansion is completed

2) After logging into the system, you can use the **df -h** command to check the size of rootfs. If it is consistent with the actual capacity of the storage device (TF card, eMMC or NVME SSD), it means that the automatic expansion is running correctly.

```
root@OpenWrt:~# df -h
Filesystem                Size      Used Available Use% Mounted on
/dev/root                14.8G     14.7G     91.6M  99% /
tmpfs                   495.5M      6.1M    489.4M   1% /tmp
tmpfs                   512.0K         0    512.0K   0% /dev
/dev/root                14.8G     14.7G     91.6M  99% /opt/docker
```

5. 4. Network port description and default network configuration

1) Orange Pi R2S has four Ethernet ports. In the OpenWRT system, the leftmost Gigabit port is configured as a WAN port by default. The remaining three ports are configured as LAN ports.



2) The WAN port is used to connect to an external network (such as a router or a higher-level network). It is configured as a DHCP client by default and will automatically obtain an IP address from the higher-level network.

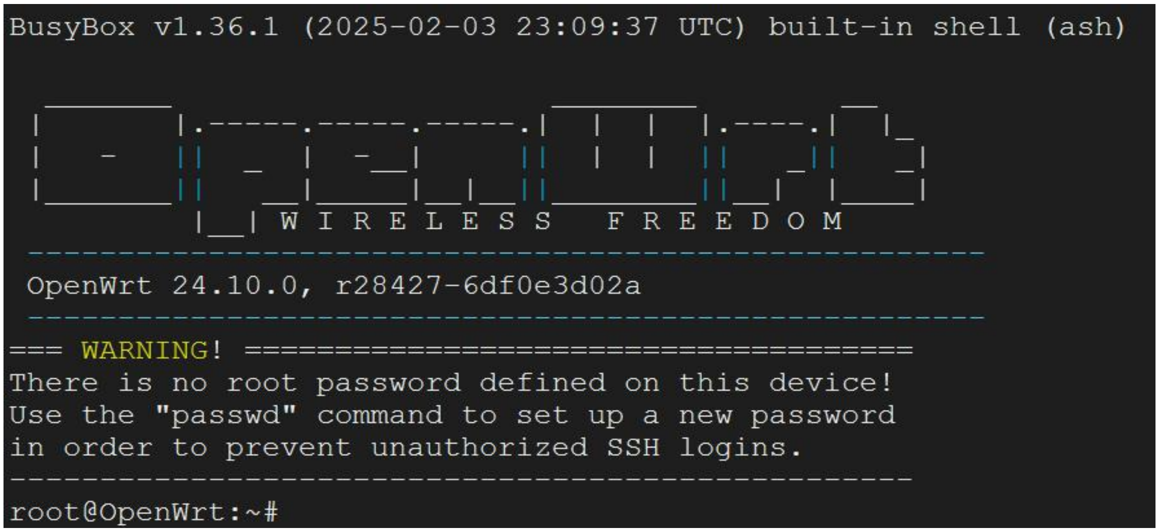
3) The LAN port is used to connect local devices, such as computers. The DHCP server is enabled by default, the network segment is **192.168.2.0/24**, and the default gateway is **192.168.2.1**.

5. 5. How to log in to the system

5. 5. 1. Login via serial port

1) First, the use of the debug serial port can refer to the section on **how to use the debug serial port**

2) The OpenWrt system will automatically log in as the **root** user by default, and the display interface is as follows



```
BusyBox v1.36.1 (2025-02-03 23:09:37 UTC) built-in shell (ash)

  _______                     ________        __
 |       |.-----.-----.-----.|  |  |  |.----.|  |_
 |   -   ||  _  |  -__|     ||  |  |  ||   _||   _|
 |_______||   __|_____|__|__||________||__|  |____|
          |__| W I R E L E S S   F R E E D O M
 -----------------------------------------------------
 OpenWrt 24.10.0, r28427-6df0e3d02a
 -----------------------------------------------------
=== WARNING! =====================================
There is no root password defined on this device!
Use the "passwd" command to set up a new password
in order to prevent unauthorized SSH logins.
--------------------------------------------------
root@OpenWrt:~#
```

5. 5. 2. Login to the system via SSH

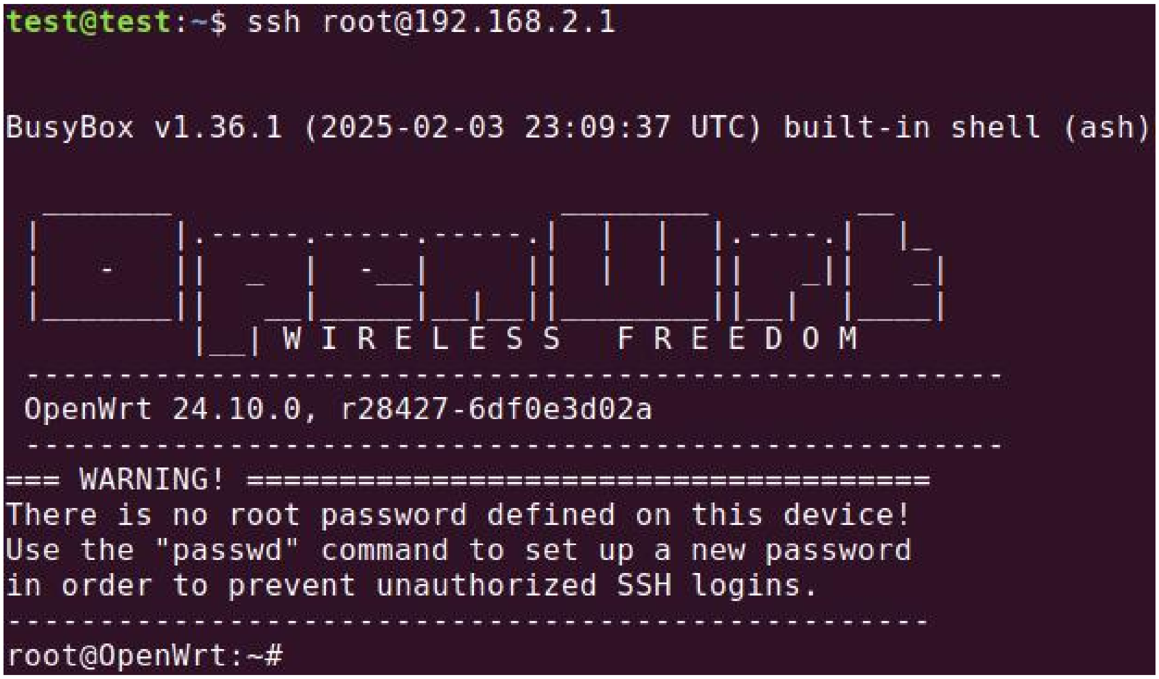
1) Use an Ethernet cable to connect the LAN port of the development board to the computer's network port. Under normal circumstances, the computer will automatically obtain an IP address from the development board through DHCP.

2) The default network segment of the LAN port in the OpenWRT system is configured as **192.168.2.0/24** (the gateway address is **192.168.2.1**), so the computer will be assigned an IP address within this network segment.

3) If the computer is installed with Ubuntu system, you can directly use SSH to log in to the board through the following command. No password is required by default.

```
test@ubuntu:~$ ssh root@192.168.2.1
```

4) After successfully logging into the system, the display is as shown below



```
test@test:~$ ssh root@192.168.2.1


BusyBox v1.36.1 (2025-02-03 23:09:37 UTC) built-in shell (ash)

  _______                     ________        __
 |       |.-----.-----.-----.|  |  |  |.----.|  |_
 |   -   ||  _  |  -__|     ||  |  |  ||   _||   _|
 |_______||   __|_____|__|__||________||__|  |____|
          |__| W I R E L E S S   F R E E D O M
 ---------------------------------------------------
 OpenWrt 24.10.0, r28427-6df0e3d02a
 ---------------------------------------------------
=== WARNING! =====================================
There is no root password defined on this device!
Use the "passwd" command to set up a new password
in order to prevent unauthorized SSH logins.
--------------------------------------------------
root@OpenWrt:~#
```

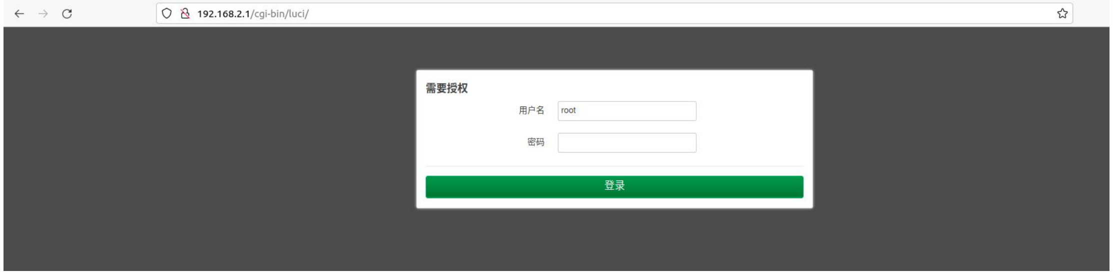
5) If the computer is installed with Windows system, you can refer to the method described in the section **"SSH remote login development board under Windows"** to log in

5. 5. 3. Log in to the LuCI management interface

1) First, use a network cable to connect the LAN port of the development board to the computer network port. Under normal circumstances, the computer will automatically obtain an IP address from the development board through DHCP.

2) The default network segment of the LAN port in the OpenWRT system is configured as **192.168.2.0/24** (the gateway address is **192.168.2.1**), so the computer will be assigned an IP address within this network segment.

3) After the computer obtains the IP address, enter the IP address of the development board LAN port **192.168.2.1** in the browser on the computer and you can see the login interface shown in the figure below.



192.168.2.1/cgi-bin/luci/

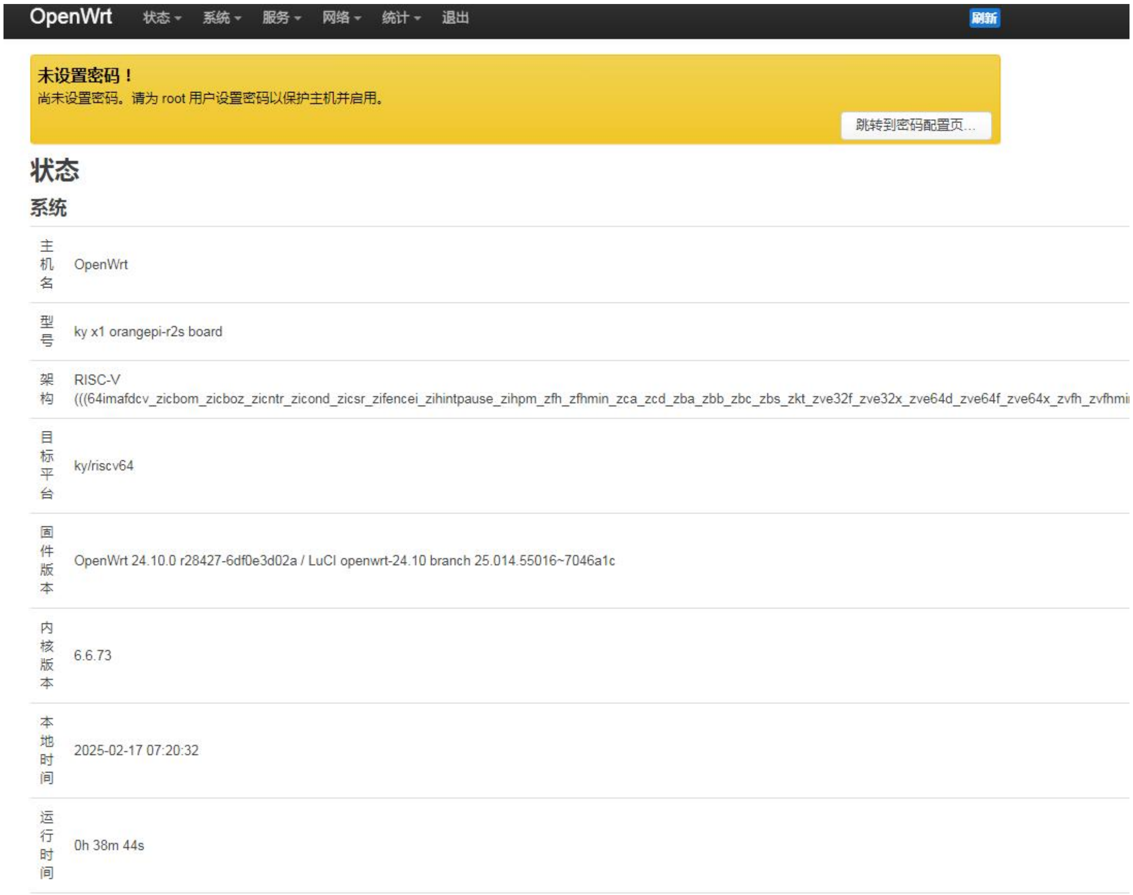
需要授权

用户名 root

密码

登录

4) **The OpenWrt system does not have a password by default**, so just click the **login button**. After a successful login, the interface is displayed as shown below



未设置密码！
尚未设置密码。请为 root 用户设置密码以保护主机并启用。

状态

系统

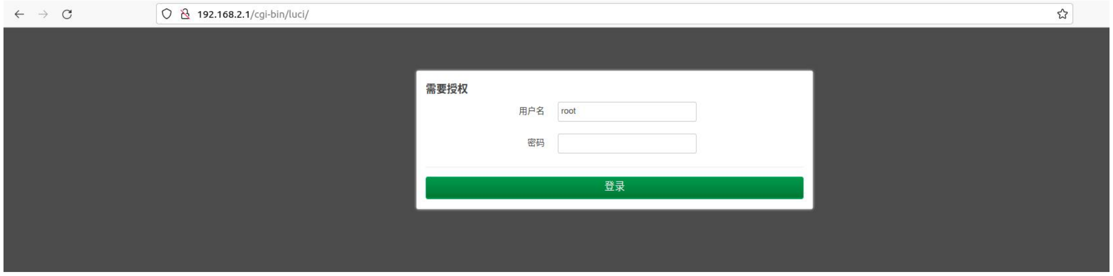
主机名	OpenWrt
型号	ky x1 orangepi-r2s board
架构	RISC-V (((64imafdcv_zicbom_zicboz_zicntr_zicond_zicsr_zifencei_zihintpause_zihpm_zfh_zfhmin_zca_zcd_zba_zbb_zbc_zbs_zkt_zve32f_zve32x_zve64d_zve64f_zve64x_zvfh_zvfhmi
目标平台	ky/riscv64
固件版本	OpenWrt 24.10.0 r28427-6df0e3d02a / LuCI openwrt-24.10 branch 25.014.55016~7046a1c
内核版本	6.6.73
本地时间	2025-02-17 07:20:32
运行时间	0h 38m 44s

5. 5. 4. Log in to the terminal through the LuCI management interface

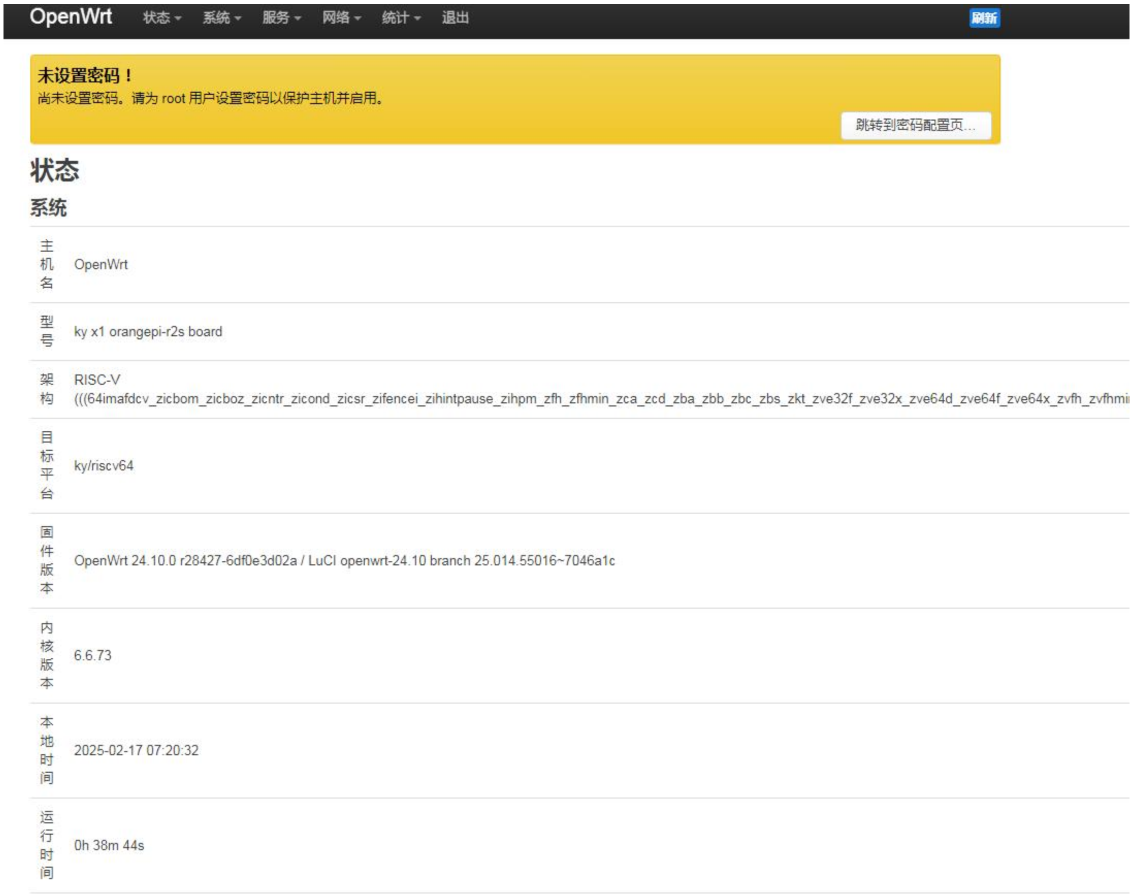
1) First, use an Ethernet cable to connect the LAN port of the development board to the computer's network port. Under normal circumstances, the computer will automatically obtain an IP address from the development board through DHCP.

2) The default network segment of the LAN port in the OpenWRT system is configured as **192.168.2.0/24** (the gateway address is **192.168.2.0/24**), so the computer will be assigned an IP address within this network segment.

3) After the computer obtains the IP address, enter the IP address of the development board LAN port **192.168.2.1** in the computer browser and you can see the login interface shown in the figure below.



5) **The OpenWrt system does not have a password by default**, so just click the **login** button. After a successful login, the interface is displayed as shown below



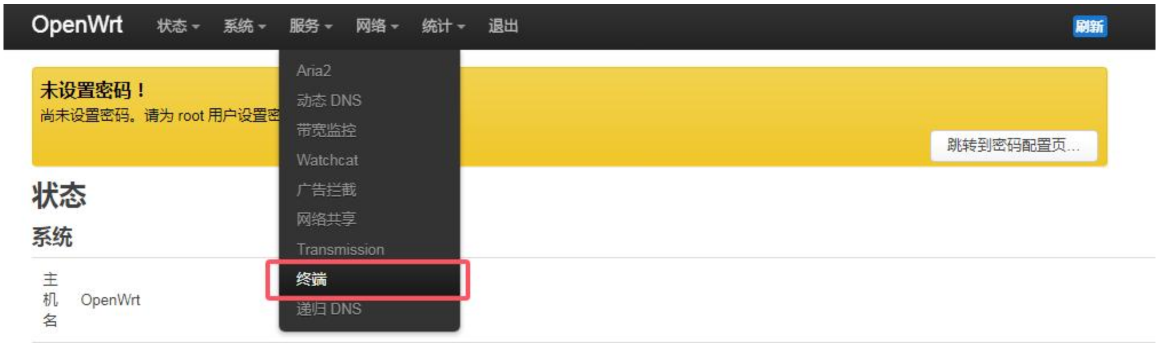
未设置密码！
尚未设置密码。请为 root 用户设置密码以保护主机并启用。

状态

系统

主机名	OpenWrt
型号	ky x1 orangepi-r2s board
架构	RISC-V (((64imafdcv_zicbom_zicboz_zicntr_zicond_zicsr_zifencei_zihintpause_zihpm_zfh_zfhmin_zca_zcd_zba_zbb_zbc_zbs_zkt_zve32f_zve32x_zve64d_zve64f_zve64x_zvfh_zvfhmi
目标平台	ky/riscv64
固件版本	OpenWrt 24.10.0 r28427-6df0e3d02a / LuCI openwrt-24.10 branch 25.014.55016~7046a1c
内核版本	6.6.73
本地时间	2025-02-17 07:20:32
运行时间	0h 38m 44s

4) Then select "**Terminal**" in the "**Service**" column of the navigation bar of the interface and click to enter



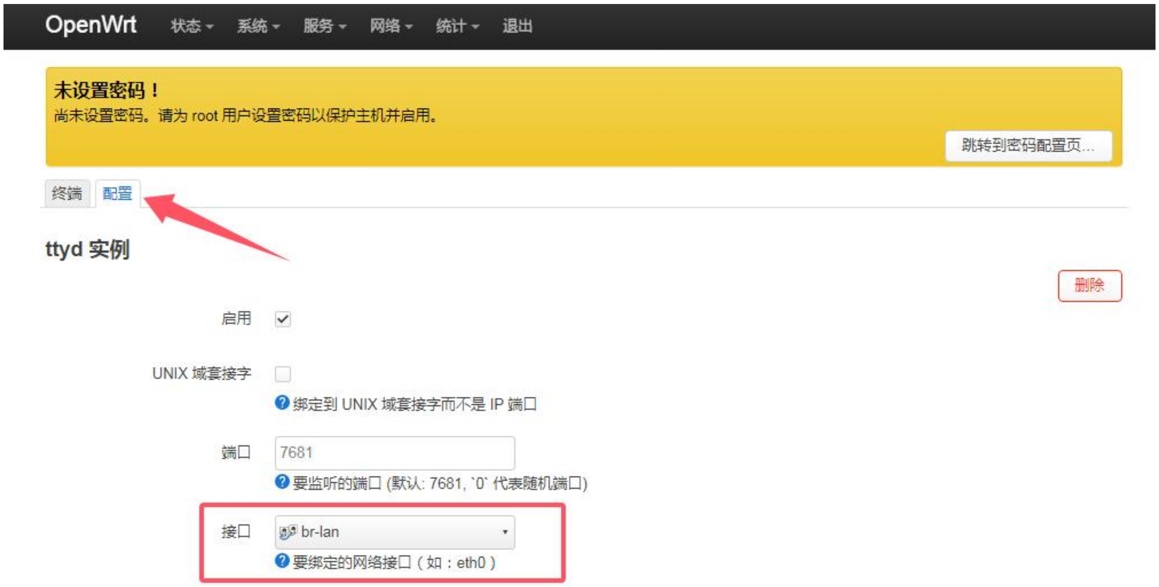
未设置密码！
尚未设置密码。请为 root 用户设置密

状态

系统

主机名 OpenWrt

5) Then click **Configure** and select **br-lan** in the Interface tab



未设置密码！
尚未设置密码。请为 root 用户设置密码以保护主机并启用。

ttyd 实例

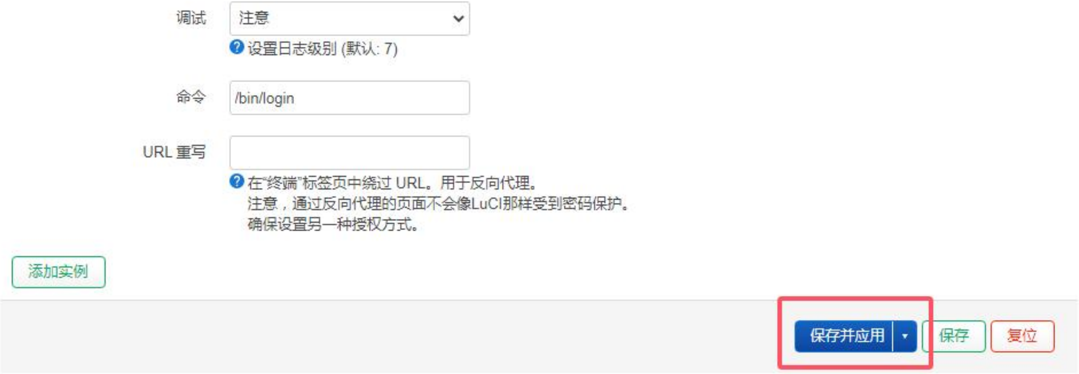
启用 ☑

UNIX 域套接字 □
绑定到 UNIX 域套接字而不是 IP 端口

端口 7681
要监听的端口 (默认: 7681, `0` 代表随机端口)

接口 br-lan
要绑定的网络接口（如：eth0）

6) Then click **Save and Apply**

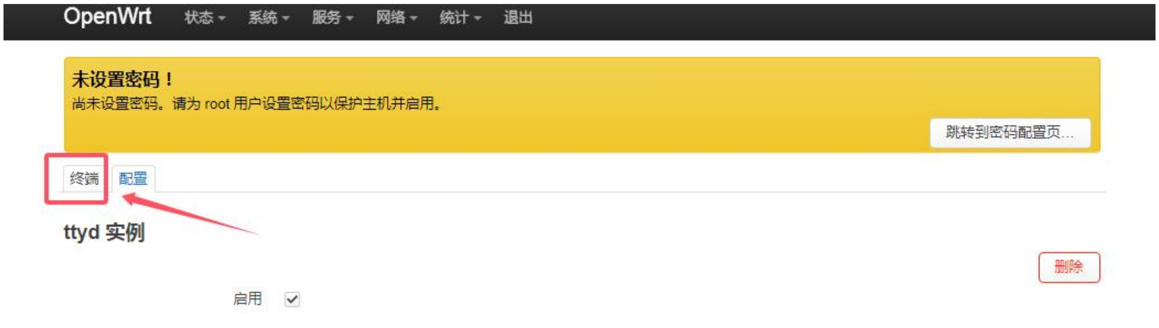


调试 注意
设置日志级别 (默认: 7)

命令 /bin/login

URL 重写
在"终端"标签页中绕过 URL。用于反向代理。
注意，通过反向代理的页面不会像LuCI那样受到密码保护。
确保设置另一种授权方式。

7) Then click **Terminal**



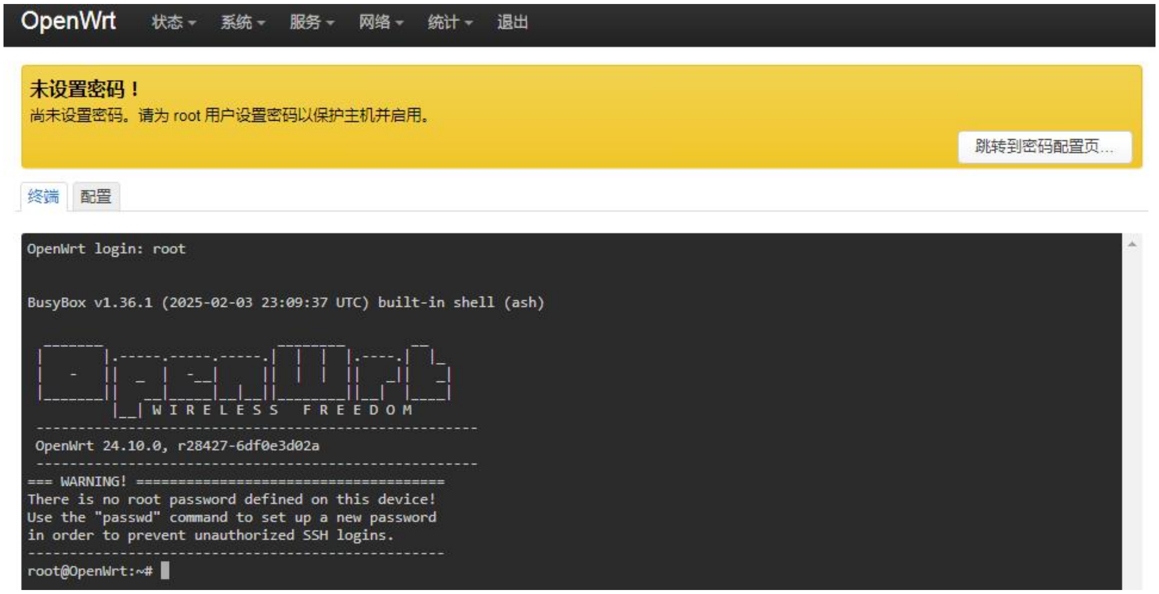
未设置密码！

尚未设置密码。请为 root 用户设置密码以保护主机并启用。

ttyd 实例

启用 ☑

8) Enter the username root to log in



未设置密码！

尚未设置密码。请为 root 用户设置密码以保护主机并启用。

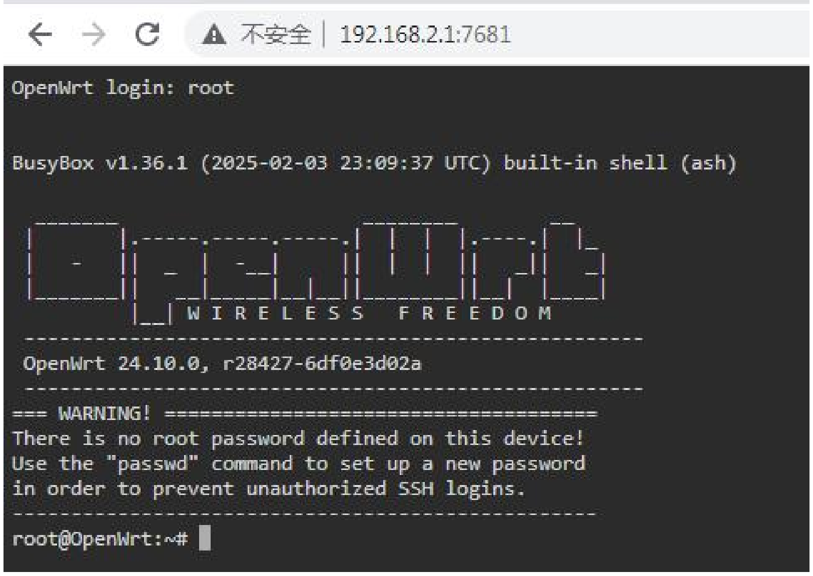
```
OpenWrt login: root


BusyBox v1.36.1 (2025-02-03 23:09:37 UTC) built-in shell (ash)

  _______                     ________        __
 |       |.-----.-----.-----.|  |  |  |.----.|  |_
 |   -   ||  _  |  -__|     ||  |  |  ||   _||   _|
 |_______||   __|_____|__|__||________||__|  |____|
          |__| W I R E L E S S   F R E E D O M
 ---------------------------------------------------
 OpenWrt 24.10.0, r28427-6df0e3d02a
 ---------------------------------------------------
=== WARNING! =====================================
There is no root password defined on this device!
Use the "passwd" command to set up a new password
in order to prevent unauthorized SSH logins.
--------------------------------------------------
root@OpenWrt:~#
```

5. 5. 5. Use IP address + port number to log in to the command line

1) First, use an Ethernet cable to connect the LAN port of the development board to the computer's network port. Under normal circumstances, the computer will automatically obtain an IP address from the development board through DHCP.

2) The default network segment of the LAN port in the OpenWRT system is configured as **192.168.2.0/24** (the gateway address is **192.168.2.1**), so the computer will be assigned an IP address within this network segment.

3) After the computer obtains the IP address, enter **192.168.2.1:7681** in the browser to log in to the OpenWRT command line.



```
OpenWrt login: root


BusyBox v1.36.1 (2025-02-03 23:09:37 UTC) built-in shell (ash)

  _______                     ________        __
 |       |.-----.-----.-----.|  |  |  |.----.|  |_
 |   -   ||  _  |  -__|     ||  |  |  ||   _||   _|
 |_______||   __|_____|__|__||________||__|  |____|
          |__| W I R E L E S S   F R E E D O M
 -----------------------------------------------------
 OpenWrt 24.10.0, r28427-6df0e3d02a
 -----------------------------------------------------
=== WARNING! =====================================
There is no root password defined on this device!
Use the "passwd" command to set up a new password
in order to prevent unauthorized SSH logins.
--------------------------------------------------
root@OpenWrt:~#
```

5. 6. How to modify the LAN port IP address through the command line

1) In the OpenWrt system, a command line tool uci is provided to easily modify, add, delete and read the contents of the configuration file. For detailed instructions, please refer to the official documentation

2) First, use the following command to obtain the network configuration. The corresponding configuration file is **/etc/config/network**. You can see that the value of **network.lan.ipaddr** is **192.168.2.1**

```
root@OpenWrt:~# uci show network
...
network.lan=interface
network.lan.device='br-lan'
network.lan.proto='static'
network.lan.ipaddr='192.168.2.1'
network.lan.netmask='255.255.255.0'
network.lan.ip6assign='60'
....
```

3) Use the following command to modify the **network.lan.ipaddr** item to a new value (**for example, 192.168.100.1**).

```
root@OpenWrt:~# uci set network.lan.ipaddr='192.168.100.1'
```

4) Then execute the submit command, and the above configuration will be written to **/etc/config/network**

```
root@OpenWrt:~# uci commit
```

5) Check the configuration file to confirm whether the modification is successful. If the IP address in red font is consistent with the one to be set, it means the modification is successful.

```
root@OpenWrt:~# cat /etc/config/network
...
config interface 'lan'
        option device 'br-lan'
        option proto 'static'
        option netmask '255.255.255.0'
        option ip6assign '60'
        option ipaddr '192.168.100.1'
...
```

6) Then restart the network through ubus to make the configuration take effect. For instructions on using ubus, please refer to the **official documentation**

```
root@OpenWrt:~# ubus call network restart
```

7) At this point, enter the command and you can see that the IP of the LAN port is already **192.168.100.1**

```
root@OpenWrt:~# ifconfig br-lan
br-lan    Link encap:Ethernet  HWaddr FE:55:13:A3:EF:E7
          inet addr:192.168.100.1  Bcast:192.168.100.255  Mask:255.255.255.0
          inet6 addr: fd60:c4cd:1033::1/60 Scope:Global
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:3 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:370 (370.0 B)
```

5. 7. How to change the root password

5. 7. 1. Modification via command line

1) First, enter passwd root in the system command line. The following prompt will appear. At this time, you can enter the password you want to set and press Enter to confirm.

```
root@OpenWrt:/# passwd root
Enter new UNIX password:
```

2) You will be prompted to re-enter your password. Enter the password again to confirm and press Enter.

```
Retype password:
```

3) The display of successful modification is as follows

```
passwd: password for root changed by root
```

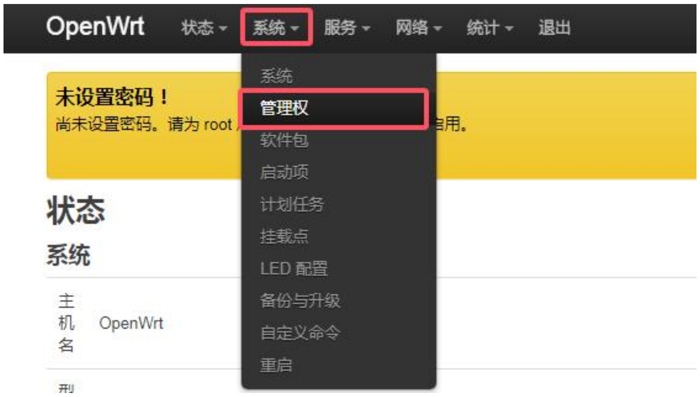
5. 7. 2. Modify through LuCI management interface

1) First refer to the **LuCI management interface** to enter the OpenWRT management interface

2) Then follow the steps below to change your password

a. Find the "**System**" option in the navigation bar and click

b. In the vertical bar options below the system, select "**Administrative Rights**" and click

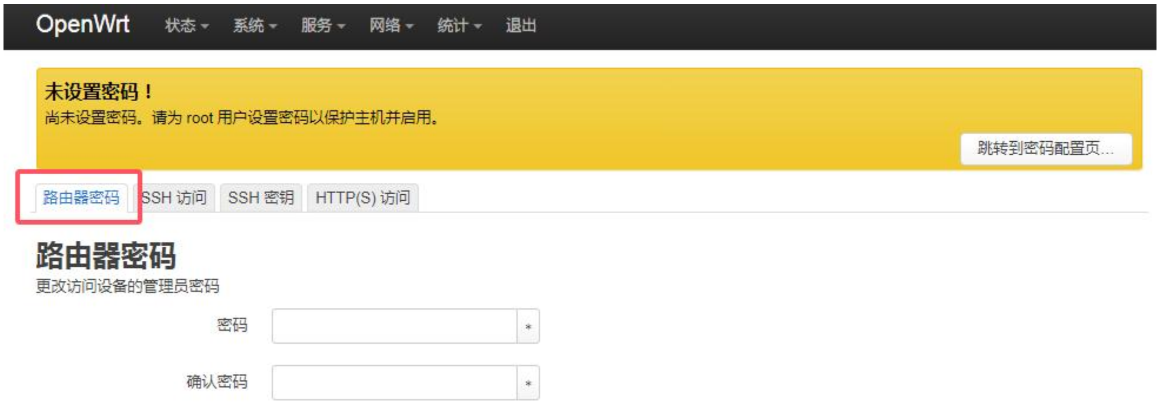


未设置密码！

尚未设置密码。请为 root

启用。

c. Select the "**Router Password**" option on the **Tab** page



未设置密码！

尚未设置密码。请为 root 用户设置密码以保护主机并启用。

路由器密码

更改访问设备的管理员密码

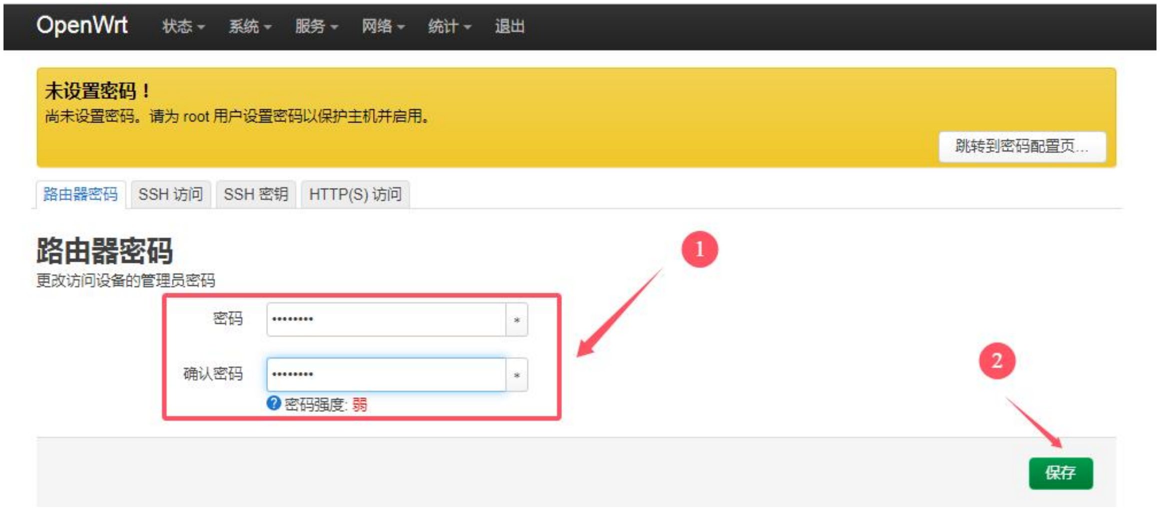
密码

确认密码

3) Modify and save the router password

a. Enter your own password in the "**Password**" and "**Confirm Password**" dialog boxes (if you are not sure whether the password is entered correctly, click the "*" icon at the back of the dialog box to display the input characters)

b. Click "**Save**" to save the newly modified password



未设置密码！

尚未设置密码。请为 root 用户设置密码以保护主机并启用。

路由器密码

更改访问设备的管理员密码

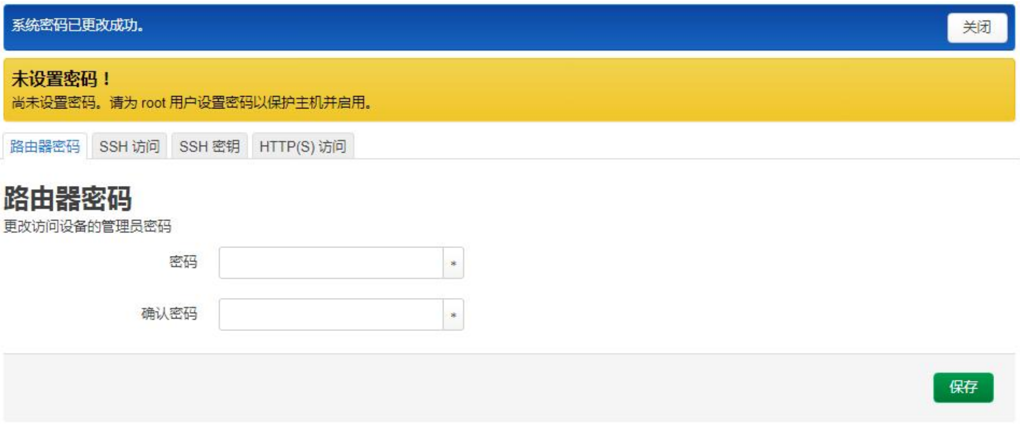
密码 ••••••••

确认密码 ••••••••

密码强度: 弱

Note: In the "Password" and "Confirm Password" dialog boxes, the passwords entered twice must be consistent.

4) After the password is successfully changed, a pop-up box will pop up saying "**System password has been changed successfully**". At this time, you need the password to log in to OpenWRT



系统密码已更改成功。

未设置密码！

尚未设置密码。请为 root 用户设置密码以保护主机并启用。

路由器密码

更改访问设备的管理员密码

密码:

确认密码:

5. 8. USB interface test

5. 8. 1. Mounting USB storage devices in the command line

1) First, insert the USB flash drive into the USB port of the Orange Pi development board.

2) Execute the following command. If you can see the output of sdX, it means that the USB disk has been successfully recognized.

```
root@OpenWrt:~# cat /proc/partitions | grep "sd*"
major minor    #blocks    name
    8              0      15126528 sda
```

3) Use the mount command to mount the USB drive to /mnt, and then you can view the files in the USB drive.

```
root@OpenWrt:~# mount /dev/sda /mnt/
root@OpenWrt:~# ls /mnt/
test.txt
```

4) After mounting, you can use the df -h command to view the capacity usage and mount point of the USB drive.

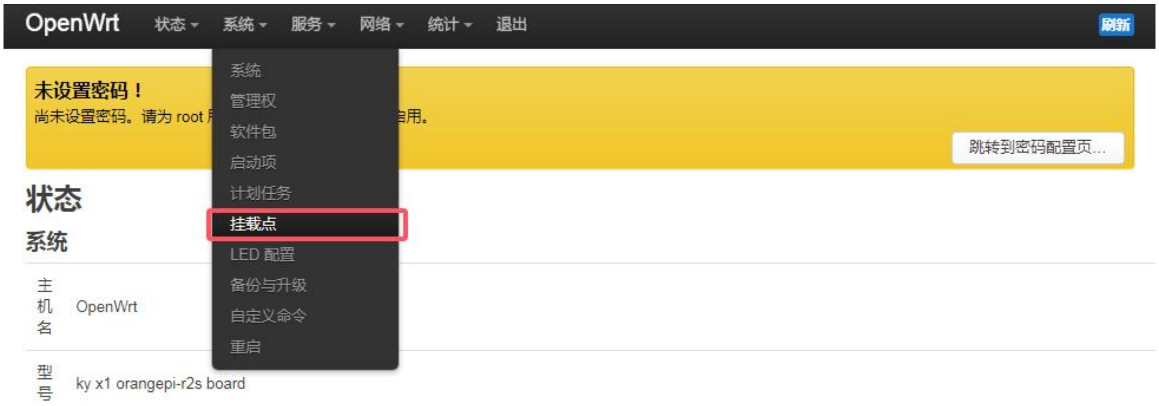
```
root@OpenWrt:~# df -h | grep "sd"
/dev/sda                14.4G       187.2M          14.2G       1% /mnt
```

5. 8. 2. Mount USB storage device in LuCI management interface

1) First connect the USB flash drive (or other storage device) to the development board via USB2.0

2) Then follow the steps to **log in to the LuCI management interface** to enter the LuCI management interface

3) Then in the LuCI management interface, click "System->Mount Point" to enter the mount point configuration interface



未设置密码！

尚未设置密码。请为 root 用 ... 启用。

状态

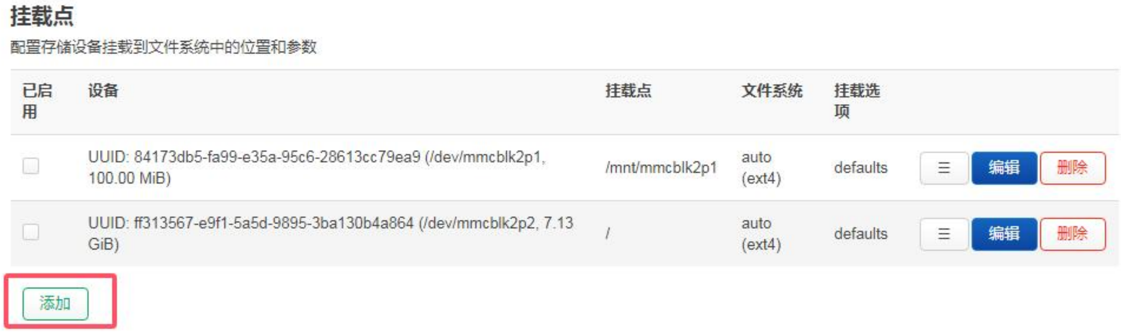
系统

主机名	OpenWrt
型号	ky x1 orangepi-r2s board

4) Then follow the steps below to add a mount point

a. Find "**Mount Point**" under the mount point **global settings interface**

b. Under the **mount point**, select the "**Add**" button and click to enter

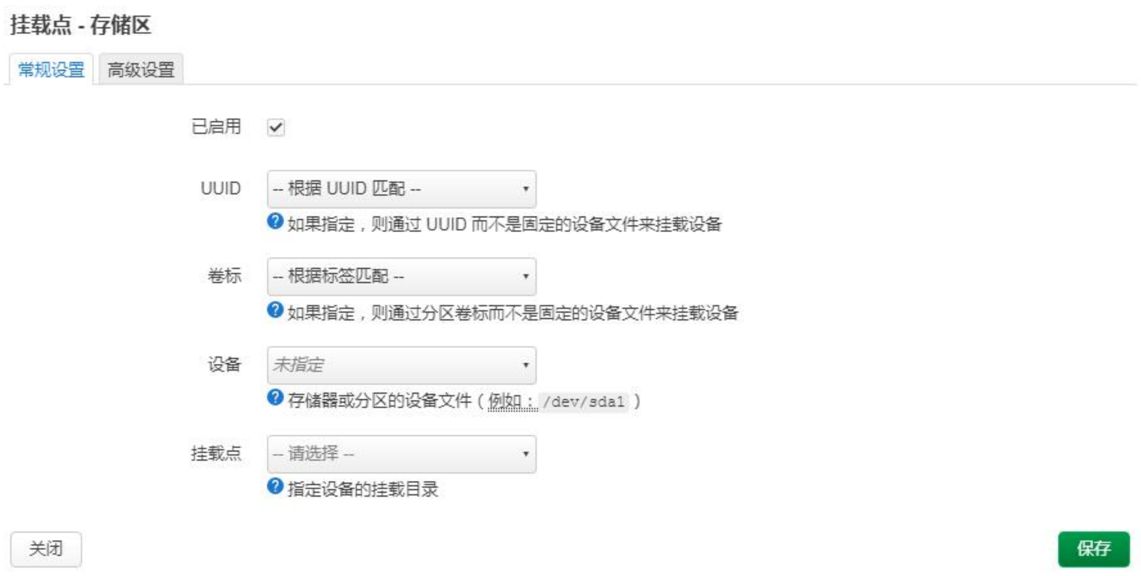


挂载点

配置存储设备挂载到文件系统中的位置和参数

已启用	设备	挂载点	文件系统	挂载选项	
□	UUID: 84173db5-fa99-e35a-95c6-28613cc79ea9 (/dev/mmcblk2p1, 100.00 MiB)	/mnt/mmcblk2p1	auto (ext4)	defaults	☰ 编辑 删除
□	UUID: ff313567-e9f1-5a5d-9895-3ba130b4a864 (/dev/mmcblk2p2, 7.13 GiB)	/	auto (ext4)	defaults	☰ 编辑 删除

c. Then the following pop-up window will pop up



挂载点 - 存储区

已启用 ☑

UUID: -- 根据 UUID 匹配 --

如果指定，则通过 UUID 而不是固定的设备文件来挂载设备

卷标: -- 根据标签匹配 --

如果指定，则通过分区卷标而不是固定的设备文件来挂载设备

设备: 未指定

存储器或分区的设备文件（例如：/dev/sda1）

挂载点: -- 请选择 --

指定设备的挂载目录

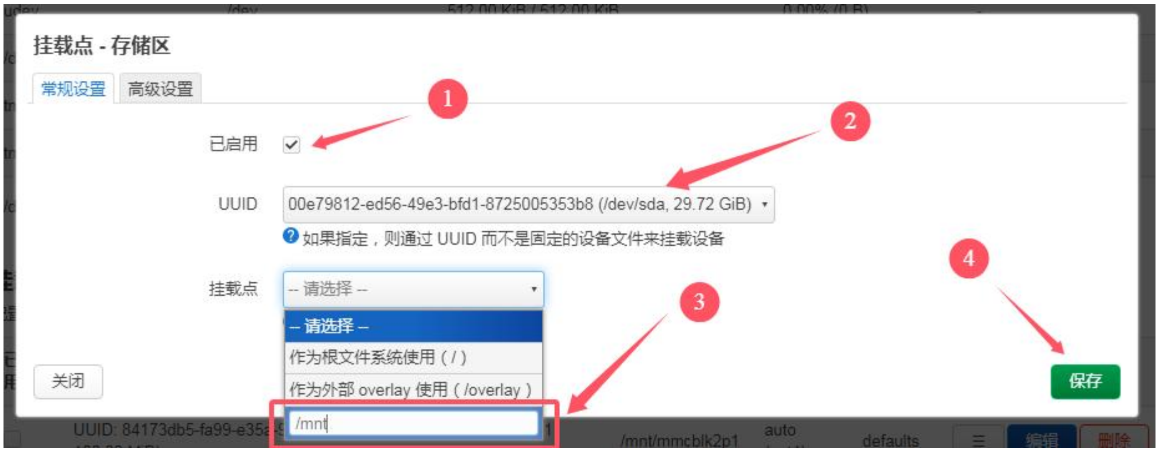
d. Then you can start mounting the storage device

a) Check "**Enabled**"

b) In the General Settings UUID column, select the actual connected device /dev/sda (select according to your own device)

c) Select "**Custom**" in the Mount Point column and fill in the target directory to be mounted. Here we take the **/mnt** directory as an example. Press Enter to confirm.

d) Then click the "**Save**" button in the lower right corner



挂载点 - 存储区

已启用 ☑

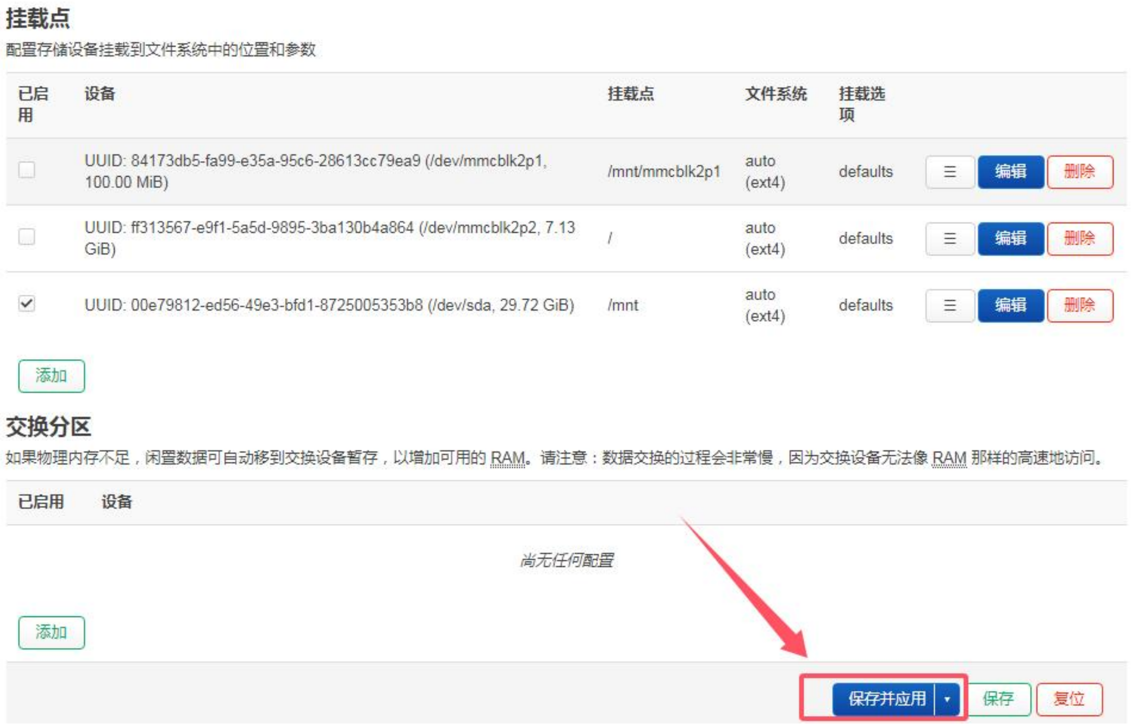
UUID: 00e79812-ed56-49e3-bfd1-8725005353b8 (/dev/sda, 29.72 GiB)

如果指定，则通过 UUID 而不是固定的设备文件来挂载设备

挂载点: -- 请选择 --

/mnt

5) Then you will return to the mount point global settings page. Click "**Save and Apply**" in the lower left corner of the page to make the mount point take effect.



挂载点

配置存储设备挂载到文件系统中的位置和参数

已启用	设备	挂载点	文件系统	挂载选项	
□	UUID: 84173db5-fa99-e35a-95c6-28613cc79ea9 (/dev/mmcblk2p1, 100.00 MiB)	/mnt/mmcblk2p1	auto (ext4)	defaults	☰ 编辑 删除
□	UUID: ff313567-e9f1-5a5d-9895-3ba130b4a864 (/dev/mmcblk2p2, 7.13 GiB)	/	auto (ext4)	defaults	☰ 编辑 删除
☑	UUID: 00e79812-ed56-49e3-bfd1-8725005353b8 (/dev/sda, 29.72 GiB)	/mnt	auto (ext4)	defaults	☰ 编辑 删除

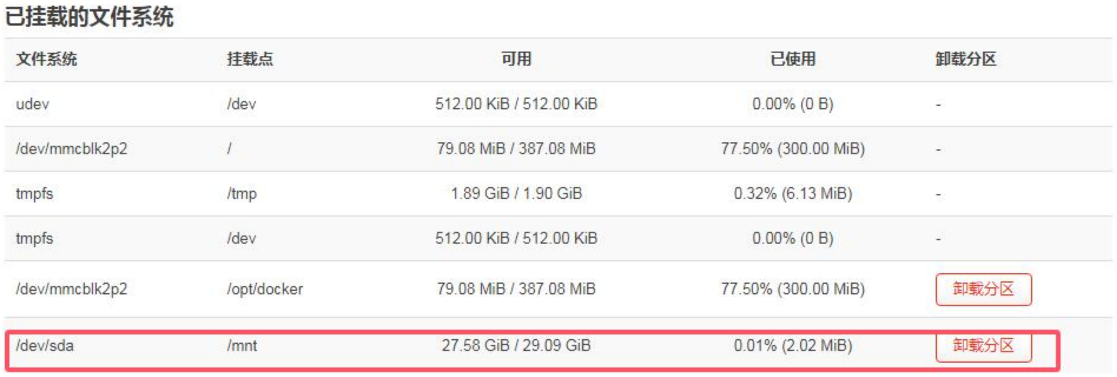
交换分区

如果物理内存不足，闲置数据可自动移到交换设备暂存，以增加可用的 RAM。请注意：数据交换的过程会非常慢，因为交换设备无法像 RAM 那样的高速地访问。

已启用	设备

尚无任何配置

6) After saving, you can see in the "**Mounted file system**" that the storage device has been mounted successfully



已挂载的文件系统

文件系统	挂载点	可用	已使用	卸载分区
udev	/dev	512.00 KiB / 512.00 KiB	0.00% (0 B)	-
/dev/mmcblk2p2	/	79.08 MiB / 387.08 MiB	77.50% (300.00 MiB)	-
tmpfs	/tmp	1.89 GiB / 1.90 GiB	0.32% (6.13 MiB)	-
tmpfs	/dev	512.00 KiB / 512.00 KiB	0.00% (0 B)	-
/dev/mmcblk2p2	/opt/docker	79.08 MiB / 387.08 MiB	77.50% (300.00 MiB)	卸载分区
/dev/sda	/mnt	27.58 GiB / 29.09 GiB	0.01% (2.02 MiB)	卸载分区

5. 9. USB wireless network card test

The currently tested USB wireless network cards are shown below. Please test other types of USB wireless network cards by yourself. If they cannot be used, you need to transplant the corresponding USB wireless network card driver.

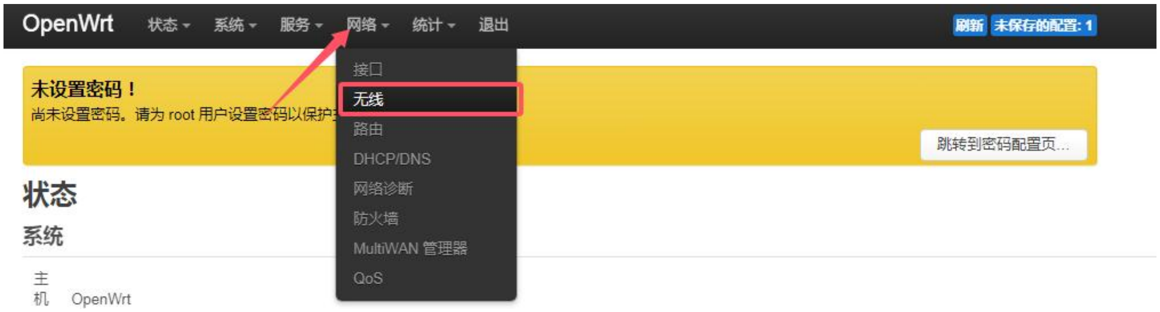
Serial number	model	

1	RTL8723BU support 2.4G WIFI+BT4.0	
2	RTL8821CU support 2.4G +5G WIFI support BT 4.2	
3	RTL8811 support 2.4G +5G WIFI	

5. 9. 1. How to connect to WIFI hotspot using USB wireless network card

1) Insert the USB wireless network card into the USB port of the development board, and then connect the power supply to power on the development board.

2) After the system is started, click **Network -> Wireless** to enter the wireless WiFi configuration interface.



未设置密码！

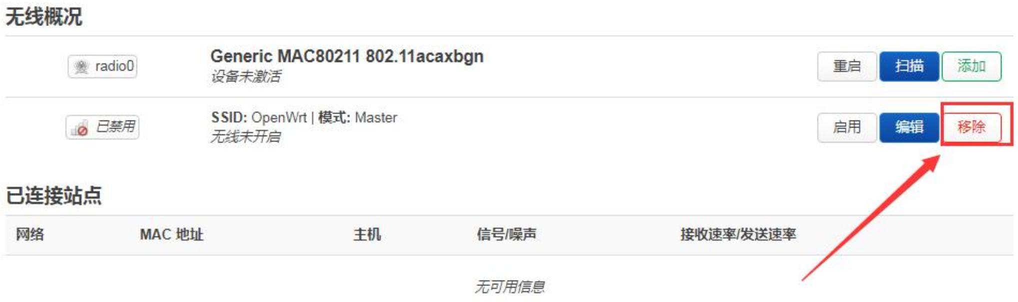
尚未设置密码。请为 root 用户设置密码以保护

状态

系统

主机 OpenWrt

3) The default wireless configuration of the OpenWRT system is **Master** mode. To facilitate the following operations, we will remove the default wireless connection.



无线概况

radio0 Generic MAC80211 802.11acaxbgn
设备未激活

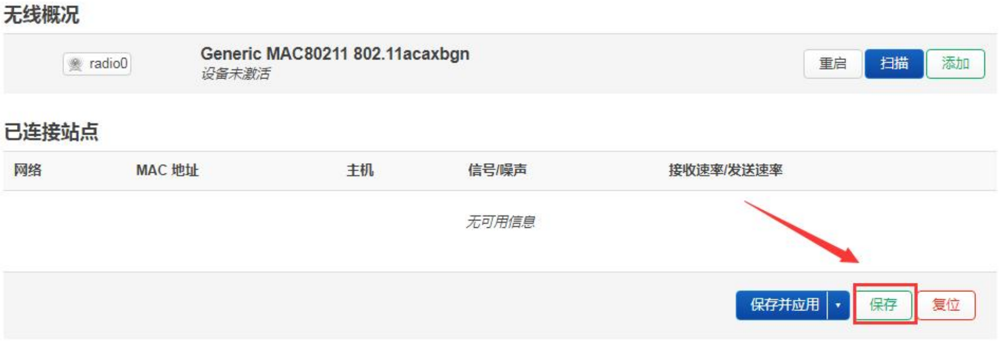
已禁用 SSID: OpenWrt | 模式: Master
无线未开启

已连接站点

网络	MAC 地址	主机	信号/噪声	接收速率/发送速率

无可用信息

4）Then click **Save** in the lower right corner of the page to make the configuration take effect.



无线概况

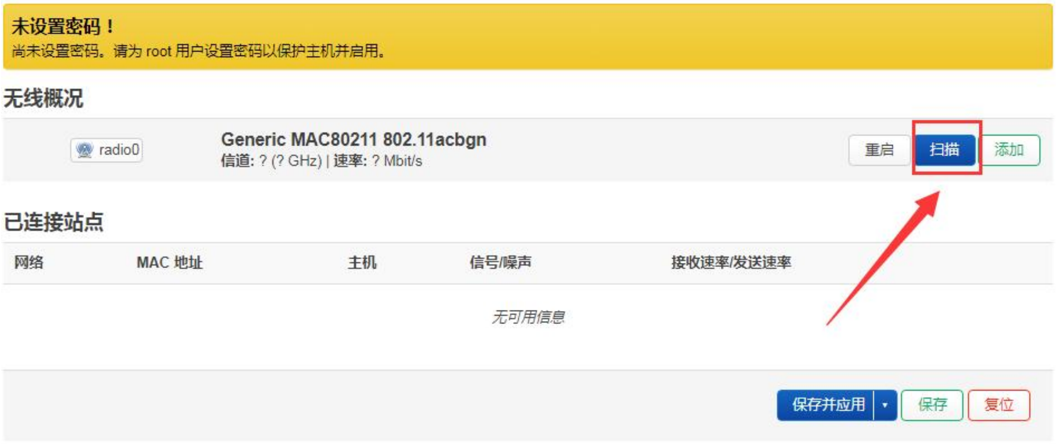
radio0 Generic MAC80211 802.11acaxbgn
设备未激活

已连接站点

网络	MAC 地址	主机	信号/噪声	接收速率/发送速率

无可用信息

5）Then click the **Scan** button to scan the surrounding WiFi hotspots.



未设置密码！
尚未设置密码。请为 root 用户设置密码以保护主机并启用。

无线概况

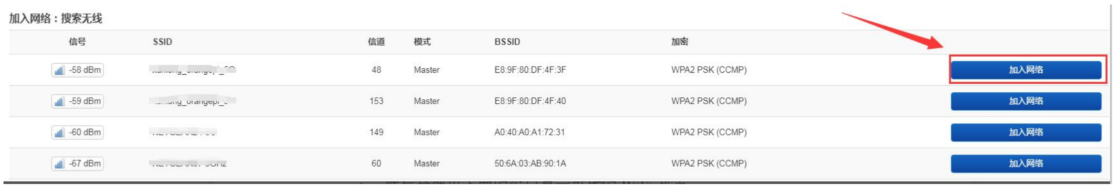
radio0 Generic MAC80211 802.11acbgn
信道: ? (? GHz) | 速率: ? Mbit/s

已连接站点

网络	MAC 地址	主机	信号/噪声	接收速率/发送速率

无可用信息

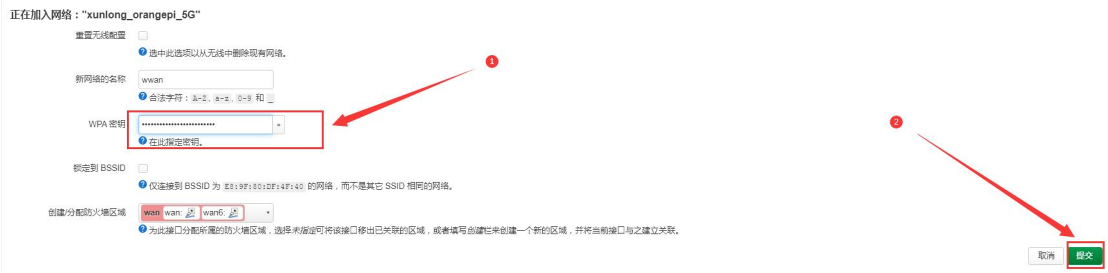
6）Then the following window will pop up showing the available WiFi hotspots. Click the **Join Network** button on the right side of the WiFi hotspot you want to connect to to connect to the WiFi hotspot.



加入网络：搜索无线

信号	SSID	信道	模式	BSSID	加密	
-58 dBm	[redacted]	48	Master	E8:9F:80:DF:4F:3F	WPA2 PSK (CCMP)	加入网络
-59 dBm	[redacted]	153	Master	E8:9F:80:DF:4F:40	WPA2 PSK (CCMP)	加入网络
-60 dBm	[redacted]	149	Master	A0:40:A0:A1:72:31	WPA2 PSK (CCMP)	加入网络
-67 dBm	[redacted]	60	Master	50:6A:03:AB:90:1A	WPA2 PSK (CCMP)	加入网络

7）Then a WiFi hotspot connection interface will pop up. Enter the hotspot password as shown in the picture below and click the **Submit** button.



正在加入网络："xunlong_orangepi_5G"

重置无线配置 □
选中此选项以从无线中删除现有网络。

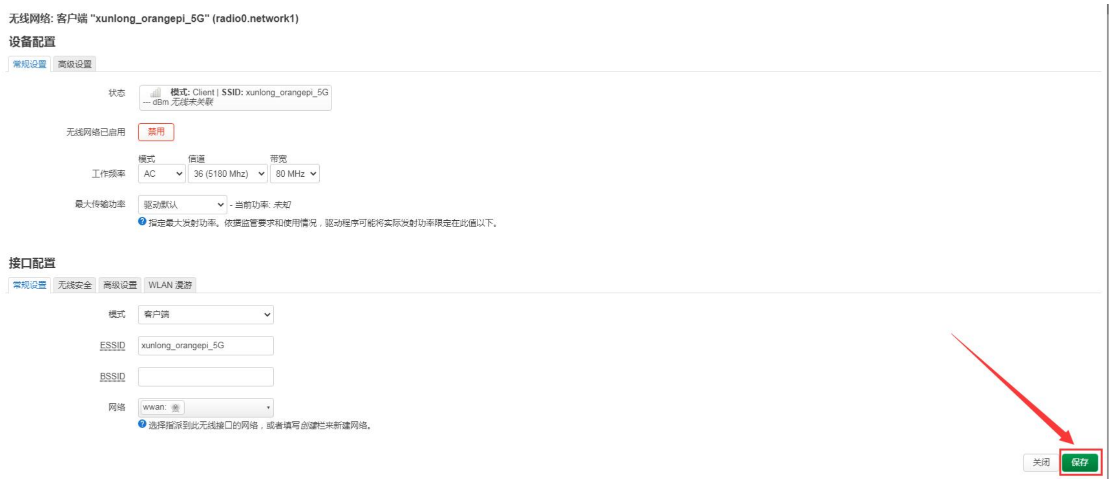
新网络的名称 wwan
合法字符：A-Z, a-z, 0-9 和 _

WPA 密钥 ••••••••••••••••••••••••••
在此指定密钥。

锁定到 BSSID □
仅连接到 BSSID 为 E8:9F:80:DF:4F:40 的网络，而不是其它 SSID 相同的网络。

创建/分配防火墙区域 wan wan: wan6:
为此接口分配所属的防火墙区域，选择未指定可将该接口移出已关联的区域，或者填写创建栏来创建一个新的区域，并将当前接口与之建立关联。

8）Then the following interface will pop up, click the **Save** button in the lower right corner.



无线网络: 客户端 "xunlong_orangepi_5G" (radio0.network1)

设备配置

状态 **模式:** Client | **SSID:** xunlong_orangepi_5G
--- dBm *无线未关联*

无线网络已启用 禁用

工作频率 模式 AC 信道 36 (5180 Mhz) 带宽 80 MHz

最大传输功率 驱动默认 - 当前功率: *未知*
指定最大发射功率。依据监管要求和使用情况，驱动程序可能将实际发射功率限定在此值以下。

接口配置

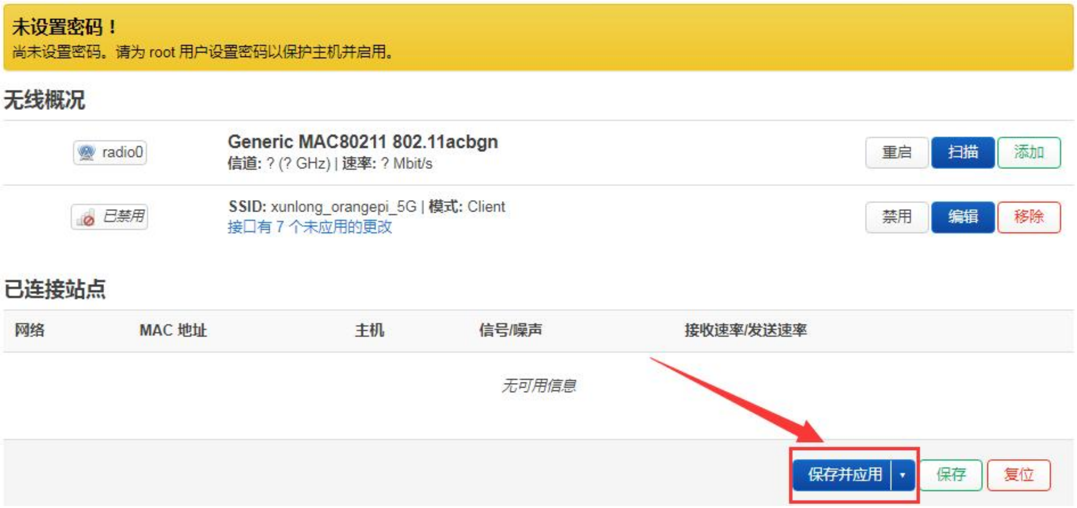
模式 客户端

ESSID xunlong_orangepi_5G

BSSID

网络 wwan:
选择指派到此无线接口的网络，或者填写*创建*栏来新建网络。

9）Finally, you will return to the main interface of wireless configuration. Click **Save and Apply** and wait for the configuration to be applied.



未设置密码！
尚未设置密码。请为 root 用户设置密码以保护主机并启用。

无线概况

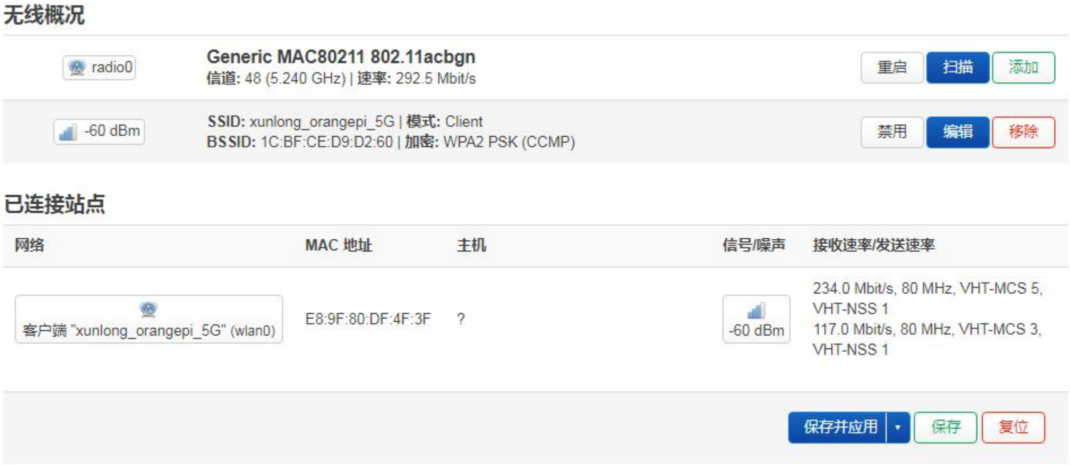
radio0	**Generic MAC80211 802.11acbgn** **信道:** ? (? GHz) | **速率:** ? Mbit/s	重启 扫描 添加
已禁用	**SSID:** xunlong_orangepi_5G | **模式:** Client 接口有 7 个未应用的更改	禁用 编辑 移除

已连接站点

网络	MAC 地址	主机	信号/噪声	接收速率/发送速率

无可用信息

10）After successfully connecting to the WiFi hotspot, the interface will be displayed as shown below.



无线概况

radio0	**Generic MAC80211 802.11acbgn** **信道:** 48 (5.240 GHz) | **速率:** 292.5 Mbit/s	重启 扫描 添加
-60 dBm	**SSID:** xunlong_orangepi_5G | **模式:** Client **BSSID:** 1C:BF:CE:D9:D2:60 | **加密:** WPA2 PSK (CCMP)	禁用 编辑 移除

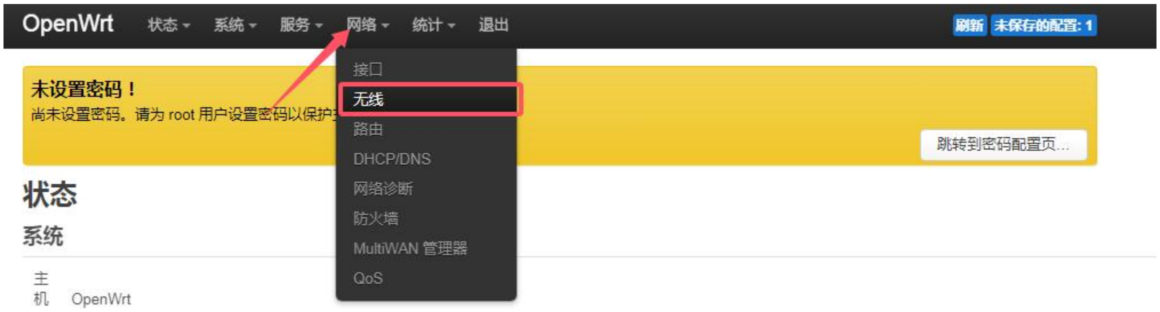
已连接站点

网络	MAC 地址	主机	信号/噪声	接收速率/发送速率
客户端 "xunlong_orangepi_5G" (wlan0)	E8:9F:80:DF:4F:3F	?	-60 dBm	234.0 Mbit/s, 80 MHz, VHT-MCS 5, VHT-NSS 1 117.0 Mbit/s, 80 MHz, VHT-MCS 3, VHT-NSS 1

5.9.2. How to create a WIFI hotspot using a USB wireless network card

1）Insert the USB wireless network card into the USB port of the development board, and then connect the power supply to power on the development board.

2）After the system is started, click **Network -> Wireless** to enter the wireless WiFi configuration interface.



未设置密码！

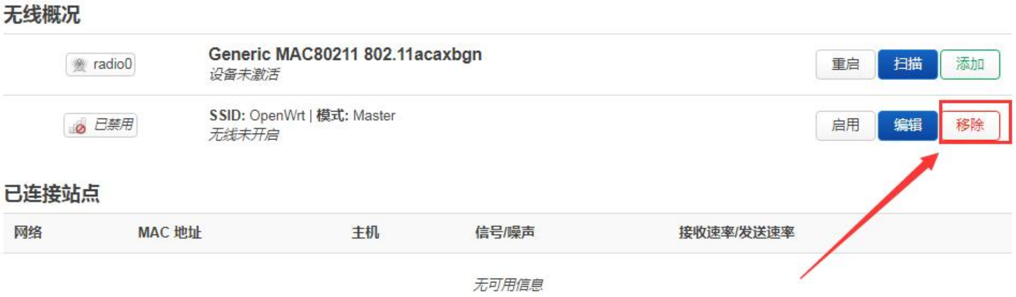
尚未设置密码。请为 root 用户设置密码以保护

状态

系统

主机 OpenWrt

3）The default wireless configuration of the OpenWRT system is **Master** mode. To facilitate the following operations, we will remove the default wireless connection.



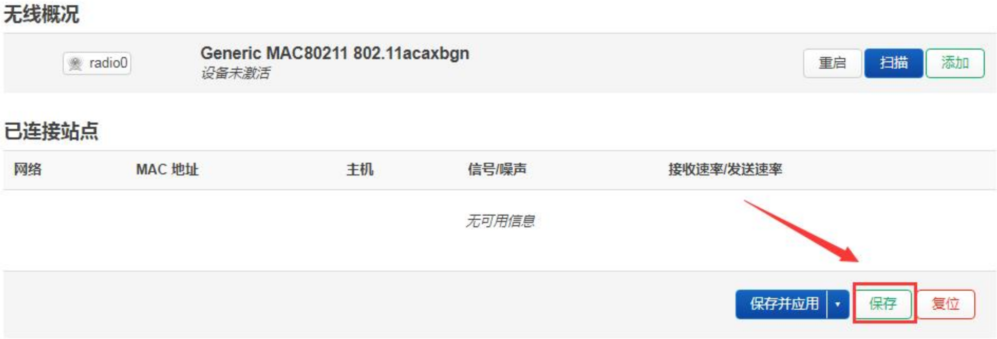
无线概况

radio0	Generic MAC80211 802.11acaxbgn 设备未激活	
已禁用	SSID: OpenWrt | 模式: Master 无线未开启	

已连接站点

网络	MAC 地址	主机	信号/噪声	接收速率/发送速率
		无可用信息		

4）Then click **Save** in the lower right corner of the page to make the configuration take effect.



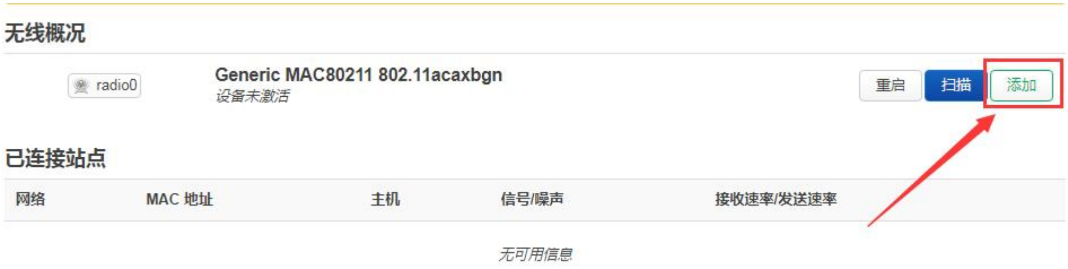
无线概况

radio0 Generic MAC80211 802.11acaxbgn
设备未激活

已连接站点

网络	MAC 地址	主机	信号/噪声	接收速率/发送速率
		无可用信息		

5）Then click the **Add** button on the right.



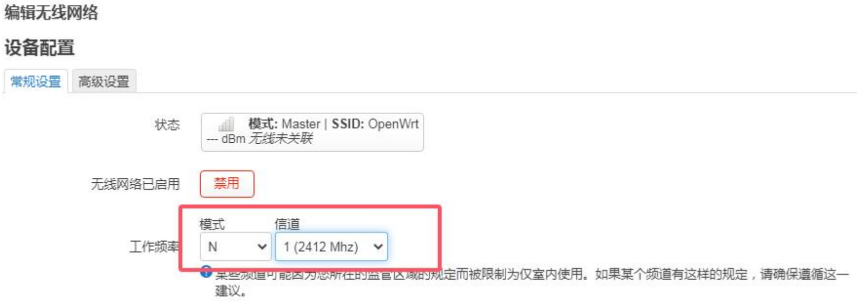
无线概况

radio0 Generic MAC80211 802.11acaxbgn
设备未激活

已连接站点

网络	MAC 地址	主机	信号/噪声	接收速率/发送速率
无可用信息				

6）In the pop-up tab **Device Configuration**, we set the parameters as shown in the figure below.



编辑无线网络

设备配置

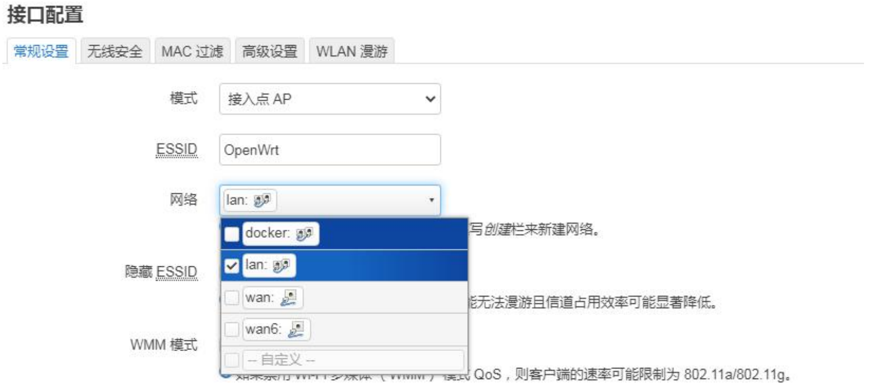
状态 **模式:** Master | **SSID:** OpenWrt
--- dBm *无线未关联*

无线网络已启用 禁用

工作频率 模式 N 信道 1 (2412 Mhz)

某些频道可能因为您所在的监管区域的规定而被限制为仅室内使用。如果某个频道有这样的规定，请确保遵循这一建议。

7）Then in **Interface Configuration** -> General Settings, set the mode to **Access Point AP**, **ESSID** (wireless network name) to **OpenWrt**, and the network to **lan**



接口配置

模式 接入点 AP

ESSID OpenWrt

网络 lan:

□ docker:
☑ lan:
□ wan:
□ wan6:
□ -- 自定义 --

写*创建*栏来新建网络。

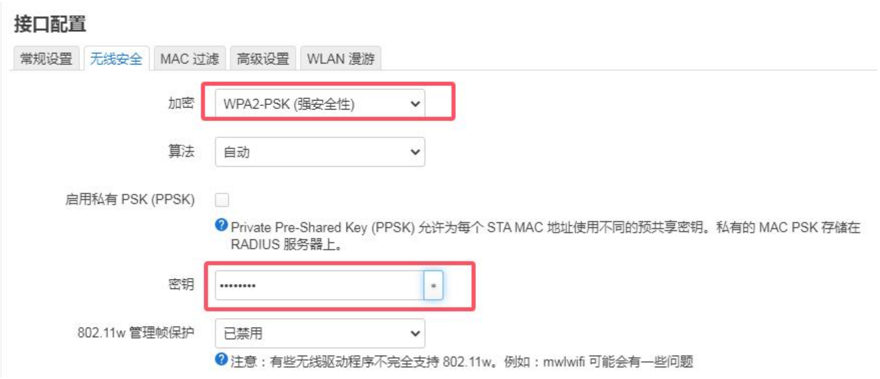
隐藏 ESSID

能无法漫游且信道占用效率可能显著降低。

WMM 模式

QoS，则客户端的速率可能限制为 802.11a/802.11g。

8）Then in **Interface Configuration -> Wireless Security**, select **WPA2-PSK** for encryption algorithm and set the key (wireless password) to **password**



接口配置

常规设置 | 无线安全 | MAC 过滤 | 高级设置 | WLAN 漫游

加密: WPA2-PSK (强安全性)

算法: 自动

启用私有 PSK (PPSK): □

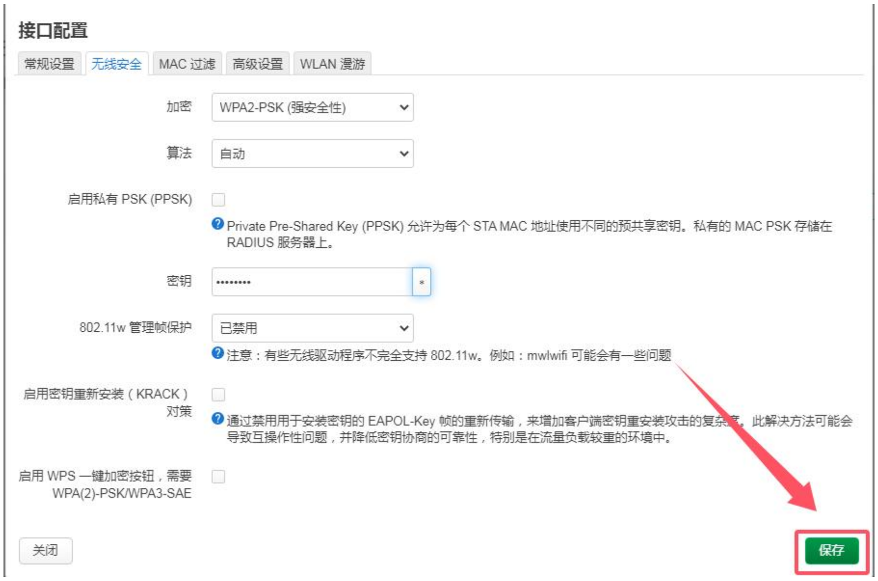
Private Pre-Shared Key (PPSK) 允许为每个 STA MAC 地址使用不同的预共享密钥。私有的 MAC PSK 存储在 RADIUS 服务器上。

密钥: ••••••••

802.11w 管理帧保护: 已禁用

注意：有些无线驱动程序不完全支持 802.11w。例如：mwlwifi 可能会有一些问题

9）After completing the above settings, click **Save** in the lower right corner of the page, and then exit the tab page



接口配置

常规设置 | 无线安全 | MAC 过滤 | 高级设置 | WLAN 漫游

加密: WPA2-PSK (强安全性)

算法: 自动

启用私有 PSK (PPSK): □

Private Pre-Shared Key (PPSK) 允许为每个 STA MAC 地址使用不同的预共享密钥。私有的 MAC PSK 存储在 RADIUS 服务器上。

密钥: ••••••••

802.11w 管理帧保护: 已禁用

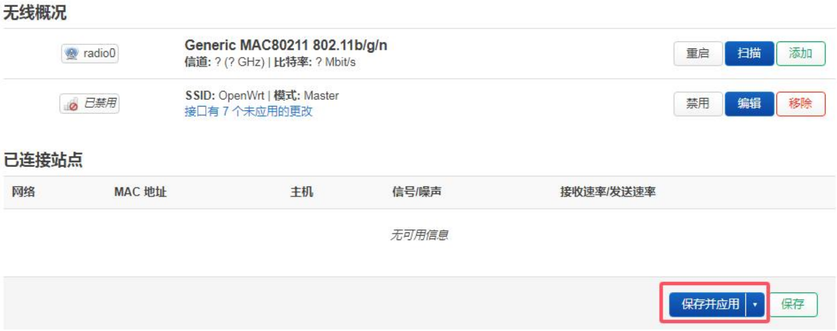
注意：有些无线驱动程序不完全支持 802.11w。例如：mwlwifi 可能会有一些问题

启用密钥重新安装（KRACK）对策: □

通过禁用用于安装密钥的 EAPOL-Key 帧的重新传输，来增加客户端密钥重安装攻击的复杂度。此解决方法可能会导致互操作性问题，并降低密钥协商的可靠性，特别是在流量负载较重的环境中。

启用 WPS 一键加密按钮，需要 WPA(2)-PSK/WPA3-SAE: □

10）Then click **Save and Apply** in the lower right corner of the page and wait for the configuration to be applied.



无线概况

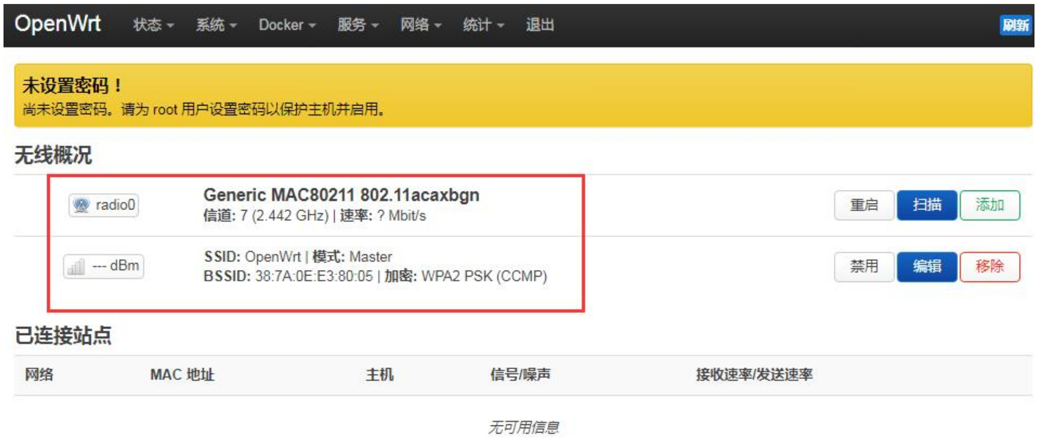
radio0 — Generic MAC80211 802.11b/g/n
信道: ? (? GHz) | 比特率: ? Mbit/s

已禁用 — SSID: OpenWrt | 模式: Master
接口有 7 个未应用的更改

已连接站点

网络	MAC 地址	主机	信号/噪声	接收速率/发送速率
无可用信息				

11）The display interface of successfully creating a hotspot is shown in the figure below



未设置密码！
尚未设置密码。请为 root 用户设置密码以保护主机并启用。

无线概况

radio0 **Generic MAC80211 802.11acaxbgn**
信道: 7 (2.442 GHz) | **速率:** ? Mbit/s

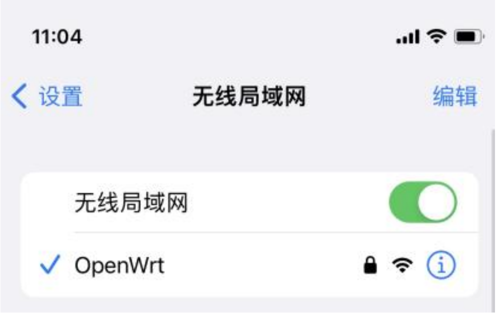
--- dBm **SSID:** OpenWrt | **模式:** Master
BSSID: 38:7A:0E:E3:80:05 | **加密:** WPA2 PSK (CCMP)

已连接站点

网络	MAC 地址	主机	信号/噪声	接收速率/发送速率

无可用信息

12）Then use your mobile phone or computer to search for the WiFi corresponding to the SSID and connect. After the connection is successful, it will be shown as follows



5. 10. Installing software packages through the command line

5. 10. 1. Installing through OPkg at the terminal

1) Update the list of available software packages

```
root@OpenWrt:/# opkg update
```

2) Get software list

```
root@OpenWrt:/# opkg list
```

3) Install the specified software package

```
root@OpenWrt:/# opkg install <Package Name>
```

4) View installed software

```
root@OpenWrt:/# opkg list-installed
```

5) Uninstall software

```
root@OpenWrt:/# opkg remove <Package Name>
```

5. 11. Installation package for OpenWRT management interface

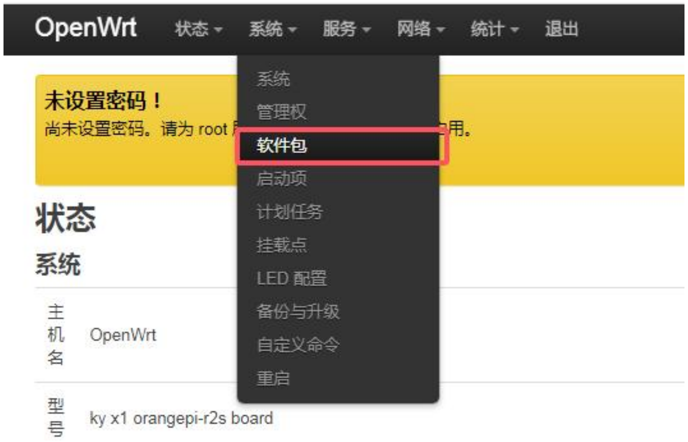
If you need to add software packages, you can install them through the OpenWRT management interface.

5. 11. 1. Viewing the List of Available Software Packages in the System

1) First, enter the software package management page

a. Find the "**System**" option in the navigation bar and click to enter

b. In the vertical bar options below the system, select "**Software Package**" and click to enter



未设置密码！
尚未设置密码。请为 root 用。

状态

系统

主机名 OpenWrt

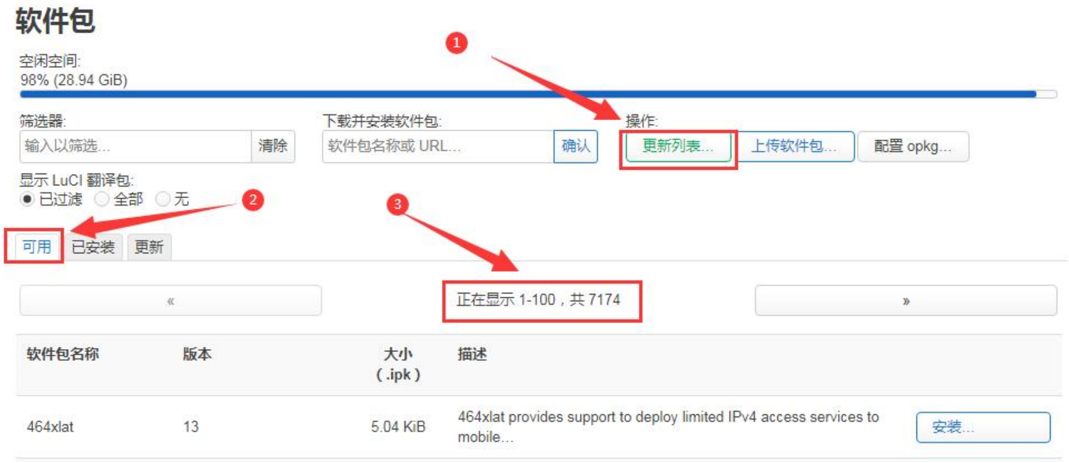
型号 ky x1 orangepi-r2s board

2) Then the main page of the software package will appear, as shown in the following figure, to obtain the list of available software

a. In the "**Operation**" option of the software package, click "**Update List**" to obtain the list of available software packages

b. On the tab page, click "**Available**" to view the currently available software packages

c. View the current number of available software packages



软件包

空闲空间:
98% (28.94 GiB)

筛选器: 输入以筛选... 清除

下载并安装软件包: 软件包名称或 URL... 确认

操作: 更新列表... 上传软件包... 配置 opkg...

显示 LuCI 翻译包:
● 已过滤 ○ 全部 ○ 无

正在显示 1-100，共 7174

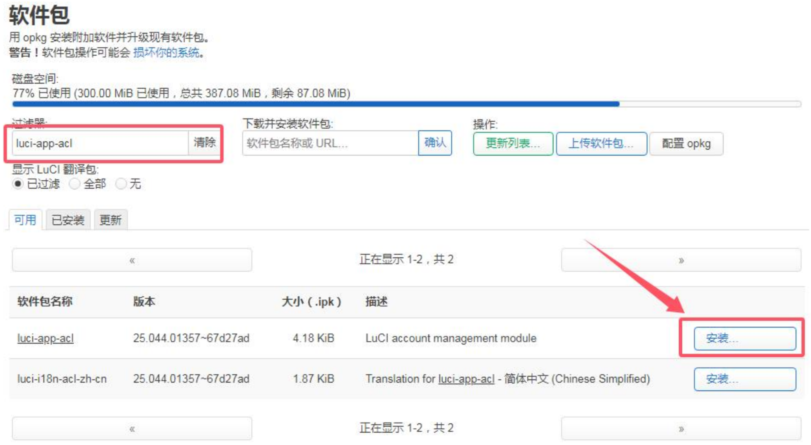
软件包名称	版本	大小（.ipk）	描述	
464xlat	13	5.04 KiB	464xlat provides support to deploy limited IPv4 access services to mobile...	安装...

5. 11. 2. Example of Installing Software Packages

1) Taking the installation of the software package "**luci-app-acl**" as an example

a. In the package management interface of OpenWRT, click on the filter dialog box and enter "**luci-app-acl**"

b. In the list of software packages, you can see the version, package size, and description information of the "**luci-app-acl**" package, and then click the "**Install**" button



软件包

用 opkg 安装附加软件并升级现有软件包。
警告！软件包操作可能会 损坏你的系统。

磁盘空间:
77% 已使用 (300.00 MiB 已使用，总共 387.08 MiB，剩余 87.08 MiB)

过滤器: luci-app-acl 清除

下载并安装软件包: 软件包名称或 URL... 确认

操作: 更新列表... 上传软件包... 配置 opkg

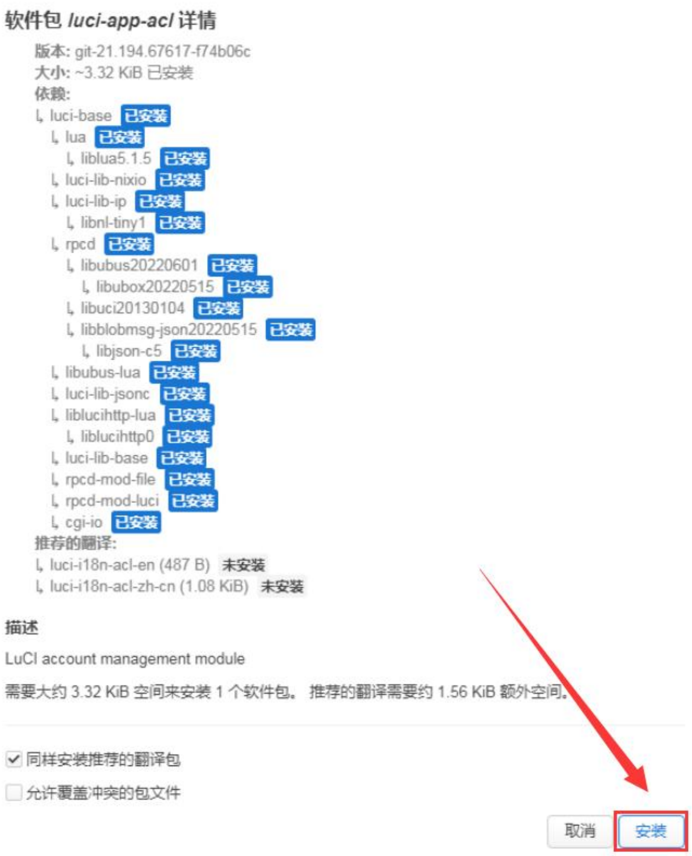
显示 LuCI 翻译包:
● 已过滤 ○ 全部 ○ 无

正在显示 1-2，共 2

软件包名称	版本	大小（.ipk）	描述	
luci-app-acl	25.044.01357~67d27ad	4.18 KiB	LuCI account management module	安装...
luci-i18n-acl-zh-cn	25.044.01357~67d27ad	1.87 KiB	Translation for luci-app-acl - 简体中文 (Chinese Simplified)	安装...

正在显示 1-2，共 2

c. Then the following pop-up window will appear, click "**Install**" to proceed



软件包 *luci-app-acl* 详情

```
版本: git-21.194.67617-f74b06c
大小: ~3.32 KiB 已安装
依赖:
↳ luci-base 已安装
  ↳ lua 已安装
    ↳ liblua5.1.5 已安装
  ↳ luci-lib-nixio 已安装
  ↳ luci-lib-ip 已安装
    ↳ libnl-tiny1 已安装
  ↳ rpcd 已安装
    ↳ libubus20220601 已安装
      ↳ libubox20220515 已安装
    ↳ libuci20130104 已安装
    ↳ libblobmsg-json20220515 已安装
      ↳ libjson-c5 已安装
  ↳ libubus-lua 已安装
  ↳ luci-lib-jsonc 已安装
  ↳ liblucihttp-lua 已安装
    ↳ liblucihttp0 已安装
  ↳ luci-lib-base 已安装
  ↳ rpcd-mod-file 已安装
  ↳ rpcd-mod-luci 已安装
  ↳ cgi-io 已安装
推荐的翻译:
↳ luci-i18n-acl-en (487 B) 未安装
↳ luci-i18n-acl-zh-cn (1.08 KiB) 未安装
```

描述

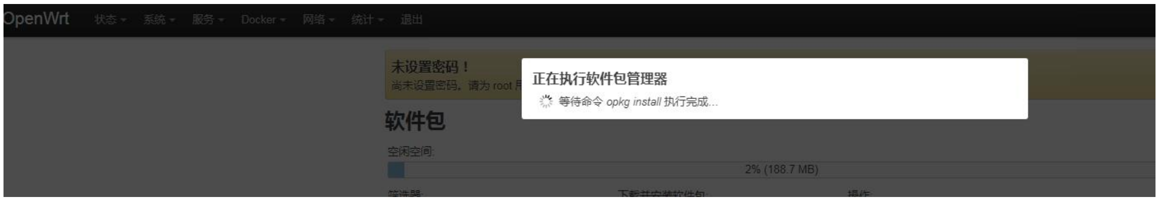
LuCI account management module

需要大约 3.32 KiB 空间来安装 1 个软件包。 推荐的翻译需要约 1.56 KiB 额外空间。

☑ 同样安装推荐的翻译包

□ 允许覆盖冲突的包文件

d. Then wait for the installation to complete



未设置密码！

尚未设置密码。请为 root 用

正在执行软件包管理器

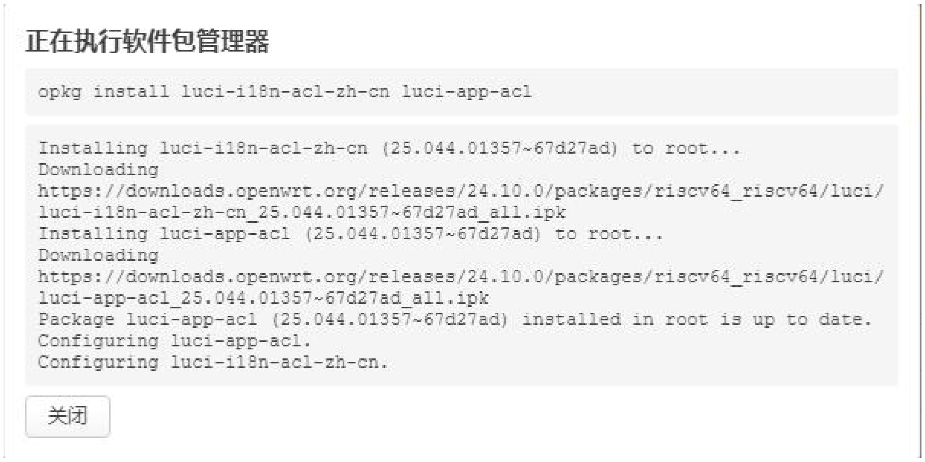
等待命令 *opkg install* 执行完成...

软件包

空闲空间:

2% (188.7 MB)

e. The display after installation is as follows



正在执行软件包管理器

```
opkg install luci-i18n-acl-zh-cn luci-app-acl
```

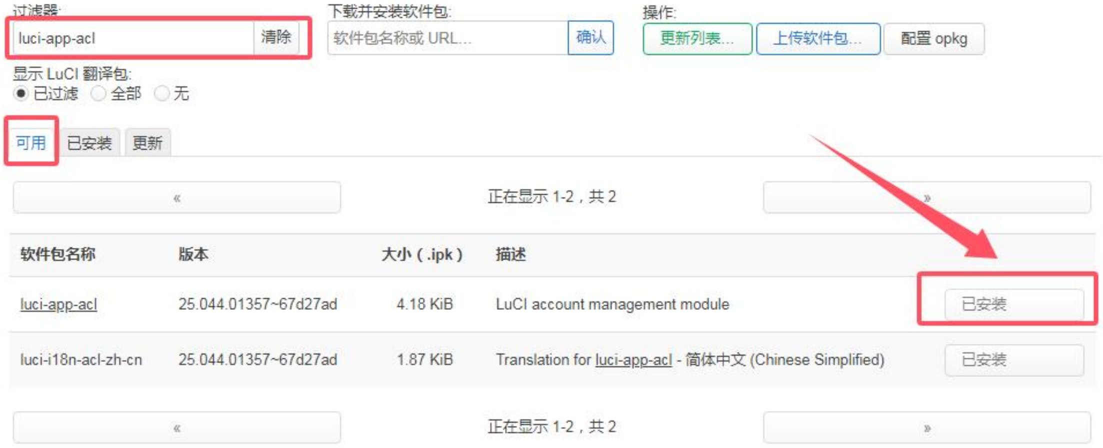
```
Installing luci-i18n-acl-zh-cn (25.044.01357~67d27ad) to root...
Downloading
https://downloads.openwrt.org/releases/24.10.0/packages/riscv64_riscv64/luci/
luci-i18n-acl-zh-cn_25.044.01357~67d27ad_all.ipk
Installing luci-app-acl (25.044.01357~67d27ad) to root...
Downloading
https://downloads.openwrt.org/releases/24.10.0/packages/riscv64_riscv64/luci/
luci-app-acl_25.044.01357~67d27ad_all.ipk
Package luci-app-acl (25.044.01357~67d27ad) installed in root is up to date.
Configuring luci-app-acl.
Configuring luci-i18n-acl-zh-cn.
```

2) Check if the software package has been successfully installed

   a. In the package management interface of OpenWRT, click on the filter dialog box and enter "**luci-app-acl**"

   b. Select and click '**Available**' on the tab page

   c. The '**luci-app-acl**' package will be displayed in the package list and updated to '**installed**' status



过滤器: luci-app-acl 清除

下载并安装软件包: 软件包名称或 URL... 确认

操作: 更新列表... 上传软件包... 配置 opkg

显示 LuCI 翻译包: ● 已过滤 ○ 全部 ○ 无

可用 已安装 更新

正在显示 1-2，共 2

软件包名称	版本	大小（.ipk）	描述	
luci-app-acl	25.044.01357~67d27ad	4.18 KiB	LuCI account management module	已安装
luci-i18n-acl-zh-cn	25.044.01357~67d27ad	1.87 KiB	Translation for luci-app-acl - 简体中文 (Chinese Simplified)	已安装

正在显示 1-2，共 2

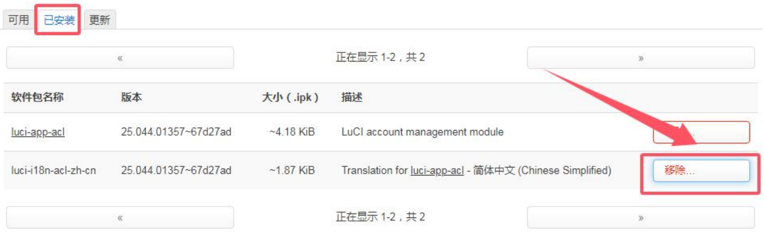
5. 11. 3. Example of Removing Software Packages

1) Taking the removal of the software package '**luci-app-acl**' as an example

   a. In the package management interface of OpenWRT, click on the filter dialog box and enter "**luci-app-acl**"

   b. Select '**Installed**' on the tab page to display a list of installed software packages

   c. Click '**Remove**' on the right to remove the corresponding software package



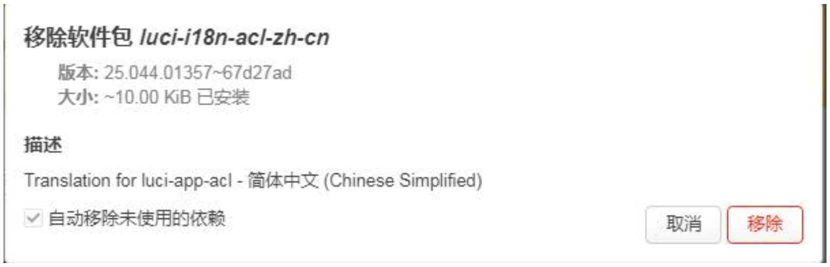
可用 已安装 更新

正在显示 1-2，共 2

软件包名称	版本	大小（.ipk）	描述	
luci-app-acl	25.044.01357~67d27ad	~4.18 KiB	LuCI account management module	
luci-i18n-acl-zh-cn	25.044.01357~67d27ad	~1.87 KiB	Translation for luci-app-acl - 简体中文 (Chinese Simplified)	移除...

正在显示 1-2，共 2

   d. Then a pop-up window will appear below, click '**Remove**' to proceed



移除软件包 *luci-i18n-acl-zh-cn*

版本: 25.044.01357~67d27ad

大小: ~10.00 KiB 已安装

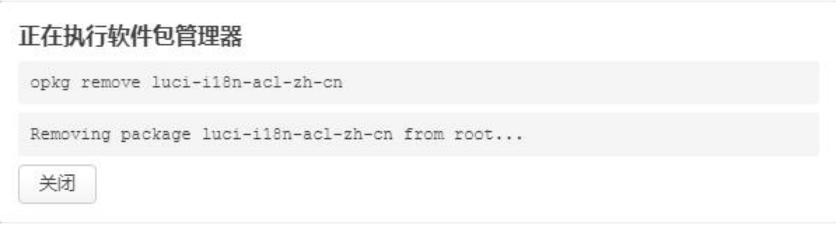
描述

Translation for luci-app-acl - 简体中文 (Chinese Simplified)

☑ 自动移除未使用的依赖

取消 移除

e. After successful removal, the display interface is as follows



正在执行软件包管理器

```
opkg remove luci-i18n-acl-zh-cn
```

```
Removing package luci-i18n-acl-zh-cn from root...
```

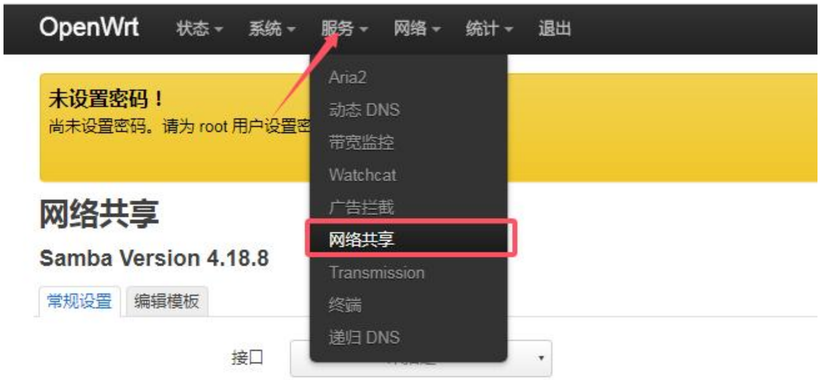
5. 12. Using Samba Network Sharing

There are two main software options for implementing OpenWRT LAN file sharing: Samba and NFS. Samba system has good compatibility, while NFS performs better. For users who need to use Windows devices, it is recommended to choose Samba.

1) Enter the Samba network share management page

a. Find the "**Services**" option in the navigation bar and click to enter

b. In the vertical bar options below the service, select "**Network Sharing**" and click to enter



未设置密码！

尚未设置密码。请为 root 用户设置密

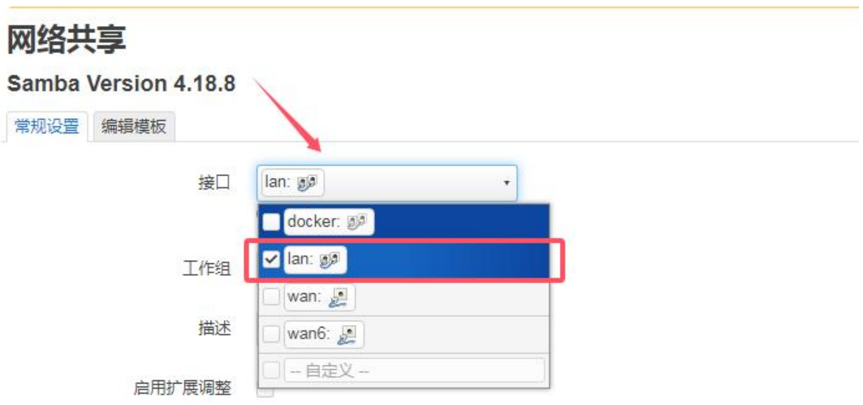
网络共享

Samba Version 4.18.8

2) Select the interface that Samba service needs to listen on

a. Select "**General Settings**" in the navigation bar of network sharing and click to enter

b. Set the interface to "**Lan**"



网络共享

Samba Version 4.18.8

接口: lan:

- ☐ docker:
- ☑ lan:
- ☐ wan:
- ☐ wan6:
- ☐ -- 自定义 --

工作组

描述

启用扩展调整

3) Set the shared directory for network sharing

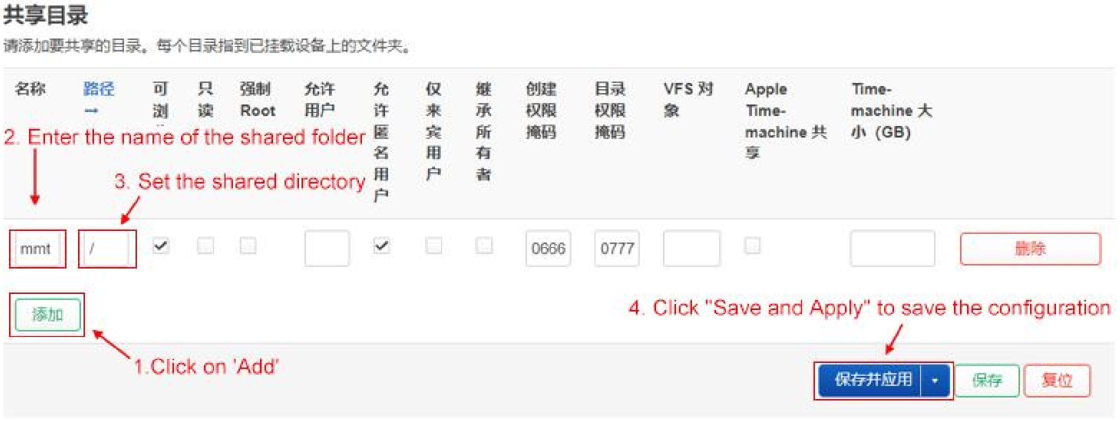
a. Click "Add" in the "**Shared Directory**" section of the "**General Settings**" for network sharing to share the directory address

b. Enter the name of the shared folder as' **mmt** 'under the name

c. Under the path of the shared directory, select the location of the shared directory "/"

d. Check '**browseable**' and '**allow anonymous users to run**'

e. Click '**Save and Apply**' to save the configuration



共享目录

请添加要共享的目录。每个目录指到已挂载设备上的文件夹。

名称	路径 →	可浏	只读	强制 Root	允许用户	允许匿名用户	仅来宾用户	继承所有者	创建权限掩码	目录权限掩码	VFS 对象	Apple Time-machine 共享	Time-machine 大小 (GB)	
mmt	/	☑	☐	☐		☑	☐	☐	0666	0777		☐		删除

2. Enter the name of the shared folder

3. Set the shared directory

1.Click on 'Add'

4. Click "Save and Apply" to save the configuration

4) Windows 10 starts network discovery and sharing

> **Note: To access Samba on the Windows 10 system, it is necessary to first confirm whether Windows 10 has started network discovery and sharing. If it has not been started, the following settings should be made first.**

a. Enable access to Samba v1/v2

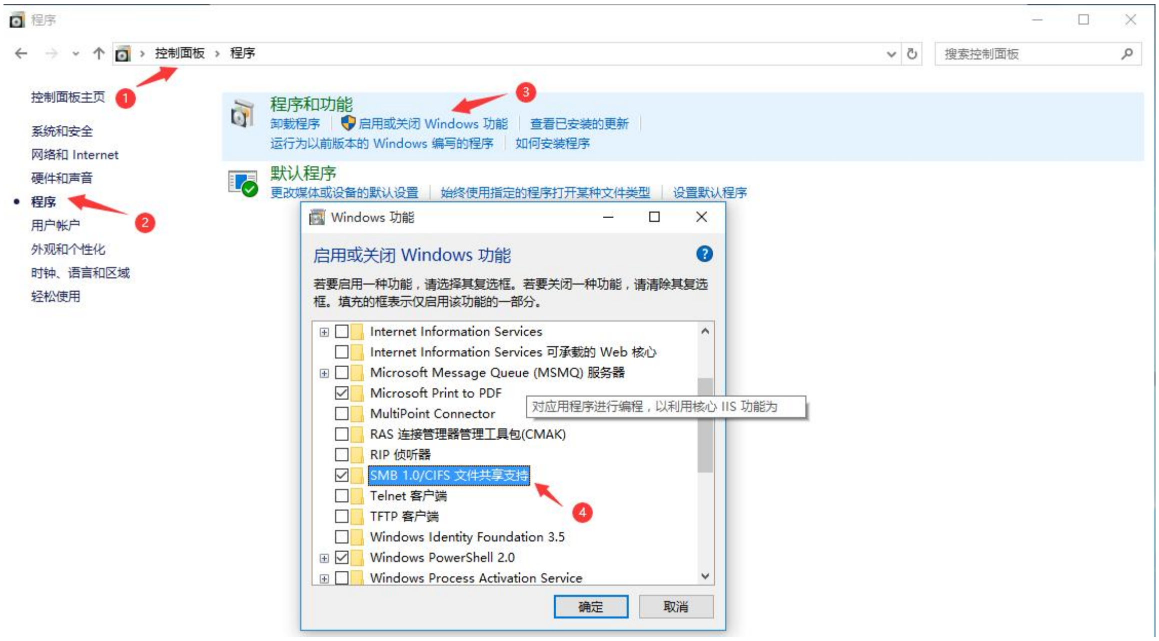
a) Enter the Control Panel of Windows 10

b) Click on "Programs" in the left navigation bar of the control panel

c) Select 'Enable or Disable Windows Features' in Programs and Features

d) Check 'SMB 1.0/CIFS file sharing support' in the pop-up box to enable or disable Windows features

e) Click 'OK' to configure the application



若要启用一种功能，请选择其复选框。若要关闭一种功能，请清除其复选框。填充的框表示仅启用该功能的一部分。

- ⊞ □ Internet Information Services
- □ Internet Information Services 可承载的 Web 核心
- ⊞ □ Microsoft Message Queue (MSMQ) 服务器
- ☑ Microsoft Print to PDF
- □ MultiPoint Connector
- □ RAS 连接管理器管理工具包(CMAK)
- □ RIP 侦听器
- ☑ SMB 1.0/CIFS 文件共享支持
- □ Telnet 客户端
- □ TFTP 客户端
- □ Windows Identity Foundation 3.5
- ⊞ ☑ Windows PowerShell 2.0
- ⊞ □ Windows Process Activation Service

b. Open Windows 10 Network Discovery

a) Enter the Control Panel of Windows 10

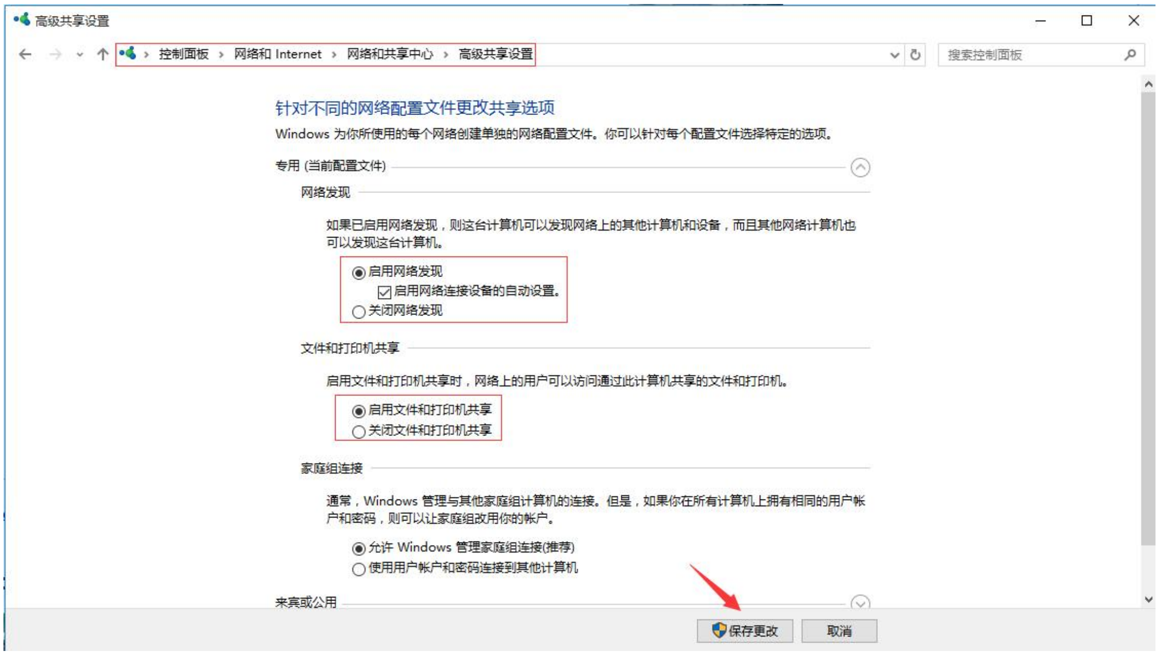
b) Select "Network and Internet" in the control panel

c) Then open the "Network and Sharing Center"

d) Click on 'Advanced Sharing Settings'

e) Open '**Enable Network Discovery**' and '**Enable File and Printer Sharing**'

f) Click 'Save Changes' to save the network discovery configuration for Windows 10



高级共享设置

控制面板 › 网络和 Internet › 网络和共享中心 › 高级共享设置

搜索控制面板

针对不同的网络配置文件更改共享选项

Windows 为你所使用的每个网络创建单独的网络配置文件。你可以针对每个配置文件选择特定的选项。

专用 (当前配置文件)

网络发现

如果已启用网络发现，则这台计算机可以发现网络上的其他计算机和设备，而且其他网络计算机也可以发现这台计算机。

● 启用网络发现
☑ 启用网络连接设备的自动设置。
○ 关闭网络发现

文件和打印机共享

启用文件和打印机共享时，网络上的用户可以访问通过此计算机共享的文件和打印机。

● 启用文件和打印机共享
○ 关闭文件和打印机共享

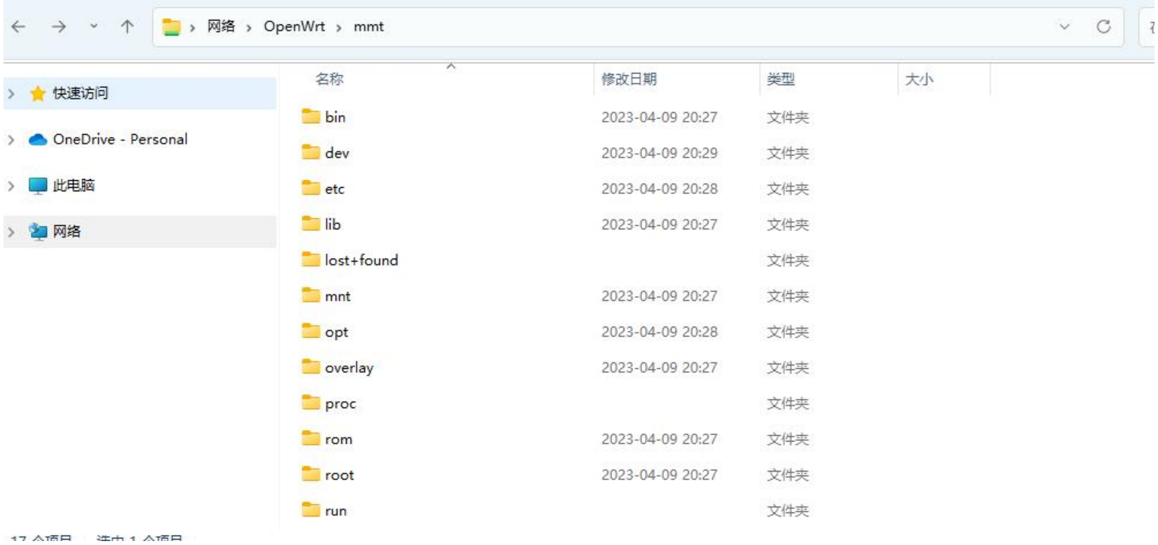
家庭组连接

通常，Windows 管理与其他家庭组计算机的连接。但是，如果你在所有计算机上拥有相同的用户帐户和密码，则可以让家庭组使用你的帐户。

● 允许 Windows 管理家庭组连接(推荐)
○ 使用用户帐户和密码连接到其他计算机

来宾或公用

5) After setting up, enter \ \ OpenWrt in the address bar of the resource manager to access the shared directory. The username is root and the password is the password set by the development board host



网络 › OpenWrt › mmt

名称	修改日期	类型	大小
bin	2023-04-09 20:27	文件夹	
dev	2023-04-09 20:29	文件夹	
etc	2023-04-09 20:28	文件夹	
lib	2023-04-09 20:27	文件夹	
lost+found		文件夹	
mnt	2023-04-09 20:27	文件夹	
opt	2023-04-09 20:28	文件夹	
overlay	2023-04-09 20:27	文件夹	
proc		文件夹	
rom	2023-04-09 20:27	文件夹	
root	2023-04-09 20:27	文件夹	
run		文件夹	

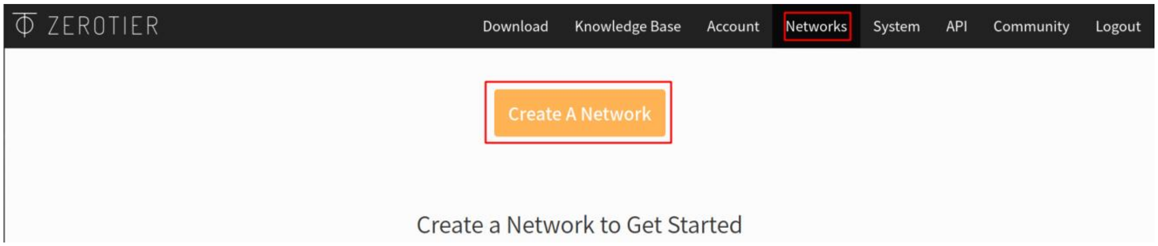
17 个项目 | 选中 1 个项目

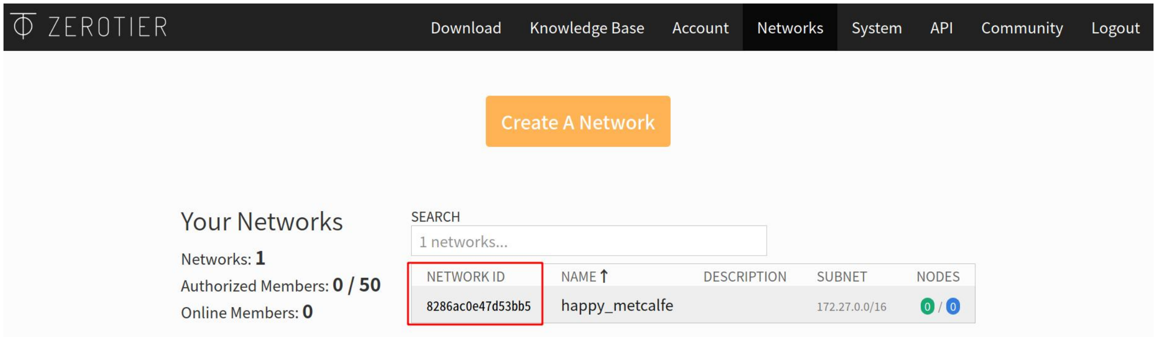
5. 13. Zerotier User Manual

The OpenWRT system has pre installed the zerotier client. After creating a

virtual LAN on the zerotier official website, the client can directly join it through the Network ID. The specific operation is shown below.

1) Log in to the zerotier official website https://my.zerotier.com/network After registering and logging in, click Network ->Create A Network to create a virtual LAN





Your Networks

Networks: **1**

Authorized Members: **0 / 50**

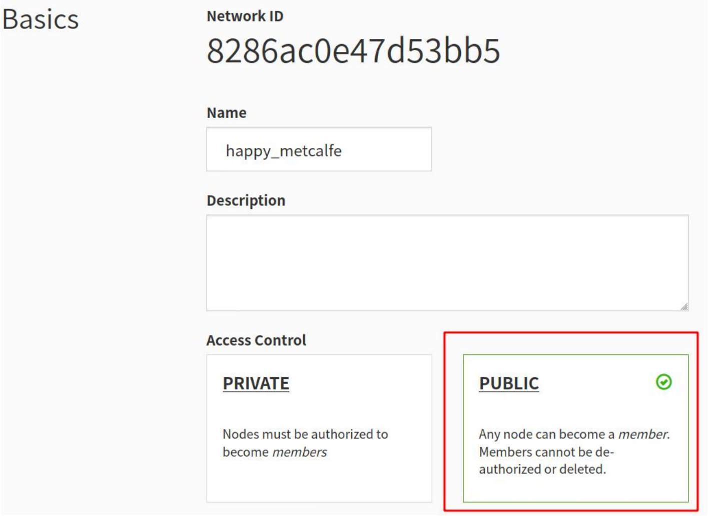
Online Members: **0**

SEARCH

1 networks...

NETWORK ID	NAME ↑	DESCRIPTION	SUBNET	NODES
8286ac0e47d53bb5	happy_metcalfe		172.27.0.0/16	0 / 0

2) Click to enter the network console page, where you can set the privacy option to public, so that network nodes that join do not need to be verified



Basics

Network ID

8286ac0e47d53bb5

Name

happy_metcalfe

Description

Access Control

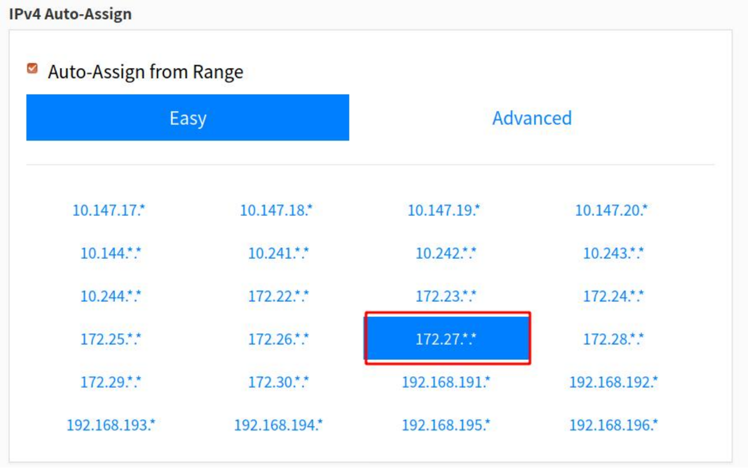
PRIVATE

Nodes must be authorized to become *members*

PUBLIC ✓

Any node can become a *member.* Members cannot be de-authorized or deleted.

3) Below, the address will be automatically assigned. Here, you can choose your own network segment, and the selected one is 172.27. **



IPv4 Auto-Assign

☑ Auto-Assign from Range

Easy	Advanced

10.147.17.*	10.147.18.*	10.147.19.*	10.147.20.*
10.144.*.*	10.241.*.*	10.242.*.*	10.243.*.*
10.244.*.*	172.22.*.*	172.23.*.*	172.24.*.*
172.25.*.*	172.26.*.*	172.27.*.*	172.28.*.*
172.29.*.*	172.30.*.*	192.168.191.*	192.168.192.*
192.168.193.*	192.168.194.*	192.168.195.*	192.168.196.*

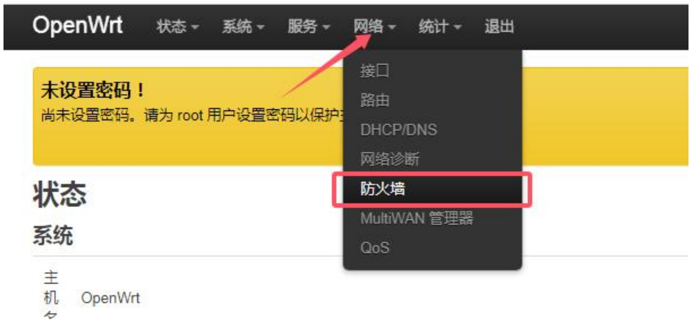
4) Enter the following command on the OpenWRT terminal to join the virtual LAN created above, **where 8286ac0e47d53bb5 is the Network ID of the virtual LAN created above**

```
root@OpenWrt:/# zerotier-one -d                          #Start the zerotier client
root@OpenWrt:/# zerotier-cli join 8286ac0e47d53bb5   #Join the network
```

5) By entering ifconfig on the terminal, it can be seen that there is already a newly added **ztks54inm2** device with an IP address of **172.27.214.213**

```
root@OpenWrt:/# ifconfig
ztks54inm2 Link encap:Ethernet    HWaddr F6:4E:DE:BF:D8:52
           inet addr:172.27.214.213    Bcast:172.27.255.255    Mask:255.255.0.0
           inet6 addr: fe80::e82f:d0ff:fe5a:867e/64 Scope:Link
           UP BROADCAST RUNNING MULTICAST    MTU:2800    Metric:1
           RX packets:18 errors:0 dropped:0 overruns:0 frame:0
           TX packets:48 errors:0 dropped:0 overruns:0 carrier:0
           collisions:0 txqueuelen:1000
           RX bytes:1720 (1.6 KiB)    TX byte81 (8.2 KiB)
```

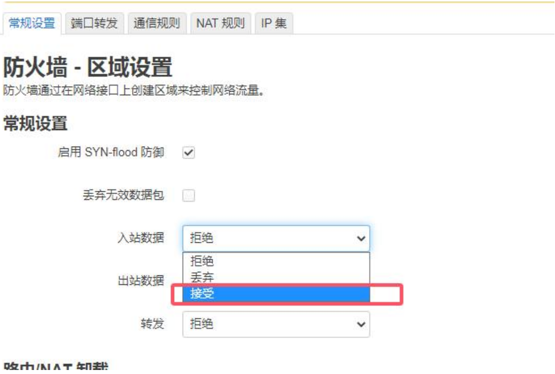
6) Then open the firewall settings in the Luci management interface



未设置密码！
尚未设置密码。请为 root 用户设置密码以保护

状态
系统
主机名 OpenWrt

7) Then the inbound data selection is accepted



防火墙 - 区域设置

防火墙通过在网络接口上创建区域来控制网络流量。

常规设置

启用 SYN-flood 防御 ☑

丢弃无效数据包 □

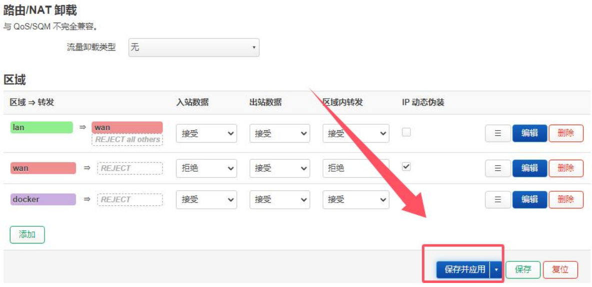
入站数据 拒绝

拒绝
丢弃
接受

出站数据

转发 拒绝

8) Then click save and apply



路由/NAT 卸载

与 QoS/SQM 不完全兼容。

流量卸载类型 无

区域

区域 ⇒ 转发	入站数据	出站数据	区域内转发	IP 动态伪装	
lan ⇒ wan REJECT all others	接受	接受	接受	□	☰ 编辑 删除
wan ⇒ REJECT	拒绝	接受	拒绝	☑	☰ 编辑 删除
docker ⇒ REJECT	接受	接受	接受		☰ 编辑 删除

9) Then install the zerotier client on another device (using Ubuntu 18.04 as an example), execute the following command to install, and restart the computer after installation is complete

```
test@ubuntu:~$ curl -s https://install.zerotier.com | sudo bash
```

10) After restarting, join the virtual LAN based on the Network ID, and you can also see that the IP address assigned by zerotier has been obtained. At this time, the Ubuntu PC and OrangePi R1 Plus LTS are in the same LAN, and they can communicate freely

```
test@ubuntu:~$ sudo zerotier-cli join 8286ac0e47d53bb5
test@ubuntu:~$ ifconfig
ztks54inm2: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>    mtu 2800
```

```
        inet 172.27.47.214   netmask 255.255.0.0   broadcast 172.27.255.255
        inet6 fe80::5ce1:85ff:fe2b:6918   prefixlen 64   scopeid 0x20<link>
        ether f6:fd:87:68:12:cf   txqueuelen 1000   (以太网)
        RX packets 0   bytes 0 (0.0 B)
        RX errors 0   dropped 0   overruns 0   frame 0
        TX packets 46   bytes 10006 (10.0 KB)
        TX errors 0   dropped 0 overruns 0   carrier 0   collisions 0
```

11) Test whether two terminals can communicate

```
root@OpenWrt:/# ping 172.27.47.214 -I ztks54inm2
PING 172.27.47.214 (172.27.47.214): 56 data bytes
64 bytes from 172.27.47.214: seq=0 ttl=64 time=1.209 ms
64 bytes from 172.27.47.214: seq=1 ttl=64 time=1.136 ms
64 bytes from 172.27.47.214: seq=2 ttl=64 time=1.203 ms
64 bytes from 172.27.47.214: seq=3 ttl=64 time=1.235 ms
^C
--- 172.27.47.214 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 1.136/1.195/1.235 ms
```

12) Zerotier other commonly used commands

```
root@OpenWrt:/# zerotier-one -d                          #Start the zerotier client
root@OpenWrt:/# zerotier-cli status                      #Obtain address and service status
root@OpenWrt:/# zerotier-cli join # Network ID           #Join the network
root@OpenWrt:/# zerotier-cli leave # Network ID          #Leave the network
root@OpenWrt:/# zerotier-cli listnetworks                #List networks
OPENWRT_DEVICE_REVISION="v0"
OPENWRT_RELEASE="OpenWrt 22.03.4 r20123-38ccc47687"
```

6. Compilation method of OpenWRT source code

6. 1. Download OpenWRT source code

1) First, execute the following command to download the openwrt-24.10 branch code

```
test@test:~$ sudo apt update
test@test:~$ sudo apt install -y git
test@test:~$ git clone https://github.com/orangepi-xunlong/openwrt.git -b openwrt-24.10
```

2) After downloading the OpenWRT code, the following files and folders will be included

```
test@test:~/openwrt$ ls
BSDmakefile  Config.in   include   Makefile  README.md  scripts  toolchain
Config  feeds.conf.default  LICENSE  package   rules.mk   target   tools
```

6. 2. Compile OpenWRT source code

1) Firstly, install the following dependency packages (currently only tested for compilation on Ubuntu 22.04. If compiling on other versions of the system, please install the dependency packages yourself according to the error message)

```
test@test:~/openwrt$ sudo apt update
test@test:~/openwrt$ sudo apt install -y ack antlr3 asciidoc autoconf   \
automake autopoint binutils bison build-essential   \
bzip2 ccache cmake cpio curl device-tree-compiler fastjar \
flex gawk gettext gcc-multilib g++-multilib git gperf haveged \
help2man intltool libc6-dev-i386 libelf-dev libglib2.0-dev \
libgmp3-dev libltdl-dev libmpc-dev libmpfr-dev \
libncurses5-dev \libncursesw5-dev libreadline-dev libssl-dev \
libtool lrzsz mkisofs msmtp nano ninja-build p7zip p7zip-full \
patch pkgconf python2.7 python3 python3-pyelftools \
libpython3-dev qemu-utils rsync scons squashfs-tools \
subversion swig texinfo uglifyjs upx-ucl unzip \
vim wget xmlto xxd zlib1g-dev
```

2) Then execute**./scripts/feeds update -a** and**./scripts/feeds install -a** download dependency package

```
test@test:~/openwrt$ ./scripts/feeds update -a
test@test:~/openwrt$ ./scripts/feeds install -a
```

3) Then choose to use the configuration file of OrangePi R2S

```
test@test:~/openwrt$ cp defconfigs/opir2s_defconfig .config
```

4) Then execute the following command to make the configuration effective

```
test@test:~/openwrt$ make defconfig
```

5) Execute the following command to start compiling the openwrt source code

```
test@test:~/openwrt$ make V=s
```

6) After compilation, the path where the image is generated is:

```
test@test:~/openwrt$ tree -L 1 bin/targets/rockchip/armv8/
bin/targets/ky/riscv64/
├── config.buildinfo
├── feeds.buildinfo
├── openwrt-ky-riscv64-x1_orangepi-r2s-ext4-sysupgrade.img.gz
├── openwrt-ky-riscv64-x1_orangepi-r2s.manifest
├── openwrt-ky-riscv64-x1_orangepi-r2s-squashfs-sysupgrade.img.gz
├── packages
├── profiles.json
├── sha256sums
├── u-boot-x1
└── version.buildinfo


2 directories, 8 files
```

7. Appendix

7. 1. User Manual Update History

Version	Date	Update Explanation
v1.0	2025-05-15	Initial version

7. 2. Image update history

Date	Update Explanation
2025-05-15	Orangepir2s_1.0.0_ubuntu_noble_server_linux6.6.63.7z openwrt-ky-riscv64-x1_orangepi-r2s-ext4-sysupgrade_20250515.img.gz openwrt-ky-riscv64-x1_orangepi-r2s-squashfs-sysupgrade_20250515.img.gz * Initial version